

Defensa contra hacke

Protección de información pr

*Este libro está dedicado
a la memoria de James Carl Coward*

Agradecimientos

La editora Diane Lowery merece la primera mención por haber sacado este libro a la luz. No sólo es una perseverante jefa de publicaciones, sino también una gran directora de desarrollo de proyectos (sus sugerencias son constantemente acertadas). Ha sido eficaz a la hora de diseñar la estructura general de este libro y ha dado unos maravillosos consejos sobre los distintos capítulos.

Malka Geffen es otra editora destacable. Ha hecho numerosas recomendaciones acertadas y prácticas a lo largo de todo el libro. Espero que se dedique de nuevo a la edición ya que los autores que consiguen trabajar con ella son afortunados.

La editora técnica Michelle Roudebush solicitó una nueva revisión cada vez que mis datos o conclusiones le parecían raros. Por supuesto, dichas revisiones siempre merecieron la pena y me evitaron más de una metedura de pata. Doy las gracias a la editora de producción Leslie Light por dirigir este proyecto de una forma tan eficiente a lo largo de todo el proceso de producción, desde el manuscrito editado hasta la maquetación de las páginas, pruebas de impresión y, finalmente, la impresión.

Por supuesto, también tengo que reconocer el magnífico trabajo de diseño de Maureen Forsy, que considero muy efectivo.

Índice

Introducción	21
--------------------	----

Los últimos peligros	21
El contenido de este libro	22
Cowboys intelectuales	23
Carnivore ataca	24
Deje de preocuparse por los virus	25
¿Hay algún secreto en este libro?	26
Cómo contactar conmigo	27

1. Peligro en Internet	29
------------------------------	----

1.1. Como la miel a las abejas	29
1.2. Sé dónde vives	30
1.3. Explorar los protocolos de Windows	32
1.4. La seguridad en Internet con Windows	33
1.4.1. Evitemos compartir archivos	34
1.5. Llamar a su propia puerta	34
1.5.1. Pruebe sus escudos y puertos	35
1.5.2. Es horrible que se difunda su información personal	35

1.5.3. Las mejores soluciones para las intentonas de los hackers	37
2. Phone Phreaks	39
2.1. ¿Quiénes son los Phone Phreaks?	40
2.1.1. Sintetizadores malvados	41
2.1.2. Bip, bip	42
3. Hackers, crackers y whackers	45
3.1. Los hacker novatos en la rampa de salida	46
3.2. Cómo hablarle a un whacker de un hacker	47
3.3. Los hacker con virus	49
3.4. Cómo enviar correo electrónico o mensajes a grupos de noticias de forma anónima	49
3.5. Hablando de correo basura: cómo deshacernos de él	50
3.5.1. Omita la dirección de correo electrónico	52
3.5.2. Disfrace su dirección de correo	53
3.5.3. Filtrar	53
3.5.4. Filtros en AOL	54
3.5.5. Luchar contra estos programas	54
3.5.6. Un aviso más	54
4. Conseguir contraseñas y hacer el Rat Dance	57
4.1. Cómo entran los hacker	58
4.1.1. Vamos a contar mentiras	58
4.1.2. Hola, soy nuevo en la oficina	60

4.1.3. El timo de la equivocación del experto	60
4.2. El problema de las contraseñas	60
4.3. Abrir el maletín misterioso	61
4.4. El Rat Dance	62
 5. La "Venus atrapamoscas" y otros anti-hacker	65
 5.1. La defensa de las empresas.....	66
5.1.1. Cebo y captura	66
5.1.2. Vigilancia constante.....	67
5.1.3. La defensa "Interfaz de 10 dedos"	67
5.2. Soluciones prácticas para empresas.....	68
5.2.1. Envíe a los Marines	70
5.2.2. Consideremos la posibilidad de un seguro	70
5.2.3. Otra vez la paradoja de las rejas en las ventanas	70
5.3. Pensar en todas las posibilidades.....	72
 6. Buscar el punto intermedio	73
 6.1. Primeros pasos para crear un lugar de trabajo seguro	73
6.1.1. Invertir la ingeniería social	73
6.1.2. Desenvolver y mantener una política de seguridad	74
6.1.3. Comprobar la identidad	76
6.1.4. Túneles, privacidad virtual y otras formas de autenticar las comunicaciones informáticas	76
6.2. Cortafuegos para todas las necesidades	77

6.2.1. Nivel a nivel	77
6.2.2. Seguridad a través de cortafuegos.....	79
6.3. Seguridad mediante cifrado	81
7. Los peligros de las conexiones de alta velocidad	83
7.1. ¿Qué hacemos?	84
7.2. Denegación de servicio	86
7.3. ¿Puede convertirse en un zombi?	87
8. Cómo proteger su banda ancha	89
8.1. Primero, seguridad.....	89
8.1.1. Cómo atraer a los hacker.....	90
8.2. Instalar un ZoneAlarm.....	90
8.2.1. Enciérrelos por completo	91
8.3. Otros cortafuegos personales	93
8.4. Hagamos una prueba	94
8.4.1. Cuidado con los PWS	94
8.4.2. ¿Hay extraños en su ordenador?	95
8.4.3. Pruebe con el escaneado gratuito de Symantec	97
8.5. Honeypots y otras tácticas.....	97
8.5.1. Pruebe con Shields Up!	97
8.6. No son precisamente las "cookies" danesas	98
8.7. Luchando contra el monstruo de las "cookies"	100
9. Privacidad en Internet	103
9.1. Ciberespionaje	104

9.1.1. Herramientas del oficio	105
9.1.1.1. Múltiples versiones de su persona	105
9.1.1.2. Cuando van cuesta abajo.....	107
9.1.1.3. Legislación futura	107
9.2. Contraatacamos	110
9.2.1. Privacidad P3P	111
9.2.2. Cuentas de correo electrónico desechables	111
9.2.3. Reenvíos anónimos (Remailers)	112
9.3. La mayor seguridad.....	113
9.4. Navegar en privado	113
9.4.1. Navegación privada con Anonymizer	114
9.4.2. Confidencialidad con libertad.....	114
9.5. También analizan su teclado	118
9.5.1. Defendámonos	119
9.5.1.1. Chequeo de Inicio	119
9.5.1.2. Buscar archivos para una palabra concreta	120
9.6. La criptografía es una potente defensa	121
10. Los elementos de la criptografía	125
10.1. Codificar versus Cifrar	125
10.2. Una antigua perfección	127
10.3. Cómo descifrar mensajes secretos	127
10.4. La gente utiliza trucos	129
10.5. El objetivo de la criptografía	130
11. El gran salto hacia adelante.....	133
11.1. La propuesta de Alberti	133

11.2. Un difícil experimento	135
11.3. La segunda gran idea de Alberti.....	136
11.3.1. Un resultado inútil	137
11.3.2. El desciframiento invierte el proceso	138
11.4. La superposición de Kerckhoffs	140
11.4.1. Construir una antitabla	141
11.4.2. Otros aspectos del polialfabeto	143
12. La llegada del ordenador	145
12.1. Velocidad y precisión perfecta	146
12.2. Algunos puntos débiles en la criptografía informática	147
12.2.1. Contraseñas incluidas	147
12.2.2. Demasiado fácil	147
12.3. Cifrado informático básico	148
12.4. Utilizar un código ya programado	148
12.5. Un defecto fatal de Xor	152
13. Infinitos monos: ataques por la fuerza bruta y otras curiosidades	153
13.1. Un problema con Xor	154
13.1.1. Un desliz fatal	157
13.1.2. El cero numérico	157
13.2. Las limitaciones de las contraseñas	158
13.2.1. Ampliar la longitud de la contraseña	160
13.2.2. Eliminar espacios	161
14. DES: un sistema público	163

14.1. Para todo el mundo	164
14.1.1. Lo que resulta extraño	165
14.1.2. El funcionamiento de DES	166
14.1.3. Los detalles técnicos	167
14.2. Desciframiento por la fuerza bruta	171
15. Claves conocidas	173
15.1. Solucionemos viejos problemas con las claves	174
15.1.1. Llévela en un maletín	174
15.1.2. Usar un centro de distribución de claves	175
15.2. La elegante solución RSA	176
15.2.1. Cifrados complejos	176
15.2.2. Fantasmas emergentes	177
15.2.3. Los números primos no son como los demás	179
15.3. Es pura matemática	181
16. Firmas electrónicas	185
16.1. Autenticación con RSA	186
16.1.1. La no repudiación	187
16.2. El Congreso se ve implicado	188
16.2.1. Surge la preocupación	189
16.2.2. El robo de identidades está a la orden del día	190
16.2.3. Combinar RSA con DES	191
17. Implementaciones criptológicas en Windows 2000	193
17.1. Los aspectos básicos de SSL	193

17.2. Certificados	194
17.3. El sistema de cifrado de archivos de Windows 2000	195
17.3.1. Es automático y transparente	195
17.3.2. Puede copiar, los demás no	196
17.3.3. Guardar archivos cifrados	197
17.3.4. Cifrar un archivo	198
17.4. Cifrar una carpeta	201
17.5. Asegurar su clave y su certificado	202
17.6. Importar un certificado .PFX y la clave	210
17.7. Agentes secretos	212
 18. Esconder datos en corrientes de fotones	213
 18.1. Criptografía atómica	213
18.1.1. Cualquier combinación posible	214
18.1.2. Se enrarece la situación	215
18.2. Mezcla de cuantos	217
18.2.1. 250 Cubits = 1 Universo	217
18.2.2. Pero son tímidos	218
18.3. Dentro de la criptografía cuántica	219
18.3.1. Alice, Bob y Eve	220
18.3.2. ¿Cómo se envía la clave?	222
 19. El perfecto sistema criptográfico indescifrable	225
 19.1. Los problemas de one time pad	226
19.2. La solución: ser aleatorias	227
19.2.1. Cuidado con Xor	227
19.2.2. Curiosidades sobre RND	228
19.2.3. Un bloc enorme	229

19.3. La historia de la máquina	230
19.3.1. Localizar la primera posición	231
19.3.2. Un mensaje estadísticamente plano	231
19.4. Trabajar la clave	232
19.5. El corazón del programa	233
19.6. Cómo utilizar el programa ROP	238
19.7. Un cifrador práctico	239
19.8. Sólo para programadores	241
19.9. Soluciones alternativas	243
 20. Se ha escapado el gran gusano	 245
20.1. La broma de los 2 millones de pesetas	247
20.2. El Gran Gusano	248
20.2.1. El daño psíquico	248
20.2.2. El pequeño programa que fue capaz	249
20.2.3. Los gusanos buenos	250
20.3. Cómo ocurrió todo	250
20.3.1. Los insectos enloquecen	251
20.3.2. Contrainoculación	252
20.3.3. Lo que salió mal	252
20.3.4. Otro intento fracasado	253
 21. Las variedades de virus: bombas lógicas, gusanos y caballos de Troya	 255
21.1. Cómo se expanden los virus	257
21.2. Huevos de pascua y bombas	258
21.2.1. Huevos inofensivos	259
21.2.2. Crucemos la puerta trasera	260

21.2.3. Cómo cargarse una base de datos	261
21.3. Caballos de Troya: nunca confíe en un griego que lleva regalos	262
21.3.1. Gusanos	262
21.3.2. Huevos inofensivos, segunda parte	263
22. Cómo Melissa cambió las reglas	265
22.1. Cómo funciona Melissa	266
22.1.1. Un ataque relativamente benigno	268
22.1.2. Lo que hace	269
22.2. Propagación por confianza	271
22.3. Medidas de seguridad que ya no funcionan	273
22.4. El virus ILOVEYOU pasa a ser el más rápido de la historia	274
22.5. Protegiéndonos	277
22.6. Desactivar Visual Basic Sripting	278
22.7. Un aviso final	279
23. Documentos que atacan (y qué puede hacer para protegerse de ellos)	281
23.1. Word nunca me haría daño	281
23.2. La mayor seguridad	283
23.2.1. Protección interna incorporada	284
23.3. Construya su propia macro	285
23.4. Disparo automático	286
23.5. ¿Es peligroso un mensaje normal de correo electrónico?	288
23.6. Actualizaciones modulares	288

23.6.1. Protegerse de objetos infectados.....	289
24. Prevención, detección y eliminación	293
24.1. Se exagera el miedo	293
24.2. Mi confesión	296
24.2.1. También Michael Jackson se resfría	297
24.2.2. La mejor protección	297
24.3. Lo que hacen los programas antivirus	298
24.3.1. Escanear las cadenas de números	299
24.3.2. Escondiéndose en el horizonte: el virus contraataca.....	299
24.3.3. La técnica Mata Hari: detección mediante atracción	301
24.3.3.1. Es fácil mantener el tamaño de un archivo	302
24.3.4. Enganchar interrupciones.....	302
24.3.5. Escaneo típico para detectar cambios en los archivos.....	302
Índice alfabético	305

Introducción

Espero que sea tan divertido leer este libro como fue para mí el proceso de investigación y posterior redacción. Mi objetivo era cubrir todos aquellos aspectos relacionados con la seguridad informática: hackers, virus y la rápida erosión de la privacidad personal.

Todos son temas fascinantes. Uno tiene la impresión de estar observando una gran partida que puede tardar décadas en terminar, si es que termina. Un hacker inteligente se lleva un punto cuando rompe las barreras de seguridad. A continuación, el contrario, que puede ser el gobierno u otros miembros del equipo anti-hackers se apunta otro tanto cuando lo atrapan, pero ya un nuevo hacker golpea de nuevo con una táctica diferente, etc. De un lado a otro, mes a mes, las fuerzas de ataque inventan nuevas formas de entrar en los sistemas protegidos, mientras las fuerzas de defensa encuentran nuevas formas para frenarlos.

He trabajado duro para conseguir que todo lo que aquí se ha escrito sea accesible al lector medio no profesional. He de reconocer que un tercio de las cuestiones tratadas son para expertos. El cifrado cuántico, por ejemplo, incluye algunos comportamientos físicos que se salen de las concepciones tradicionales, siendo sinceros, bastante escalofriantes. Sin embargo, he intentado proporcionar descripciones de fácil comprensión y ejemplos que aclaran los aspectos más técnicos tratados a lo largo de todo el libro.

Los últimos peligros

Cualquiera que use Internet, está conectando su ordenador a una inmensa red. Así, su disco duro queda expuesto a

la entrada de intrusos y sus movimientos, a los fisgones. La mayoría de la gente que navega es inofensiva pero también están esos *otros*.

A algunos de los otros les gusta dejar sorpresas desagradables en su ordenador (virus, bombas lógicas, gusanos y otras pestes). A otros les gusta dañar su disco duro y destruir su información.

Otros sólo quieren observarle: graban su correo electrónico, le echan un vistazo a sus cuentas, intentan ver cómo se comporta y, en algunos casos, llegan a apropiarse de su identidad y lanzarse a comprar como locos.

Por supuesto, a mí me gusta comprar a lo loco alguna vez como a cualquier otro. Pero, al contrario que nosotros, los hackers utilizan tarjetas de crédito ajenas para hacerlo. Puede llegar a ser sorprendentemente fácil robar la identidad de una persona. A pesar de que las pérdidas económicas no tienen por qué ser cuantiosas, después de un robo de identidad, tendrá problemas para recuperar su credibilidad.

Muchos de los hackers son totalmente inofensivos. Un famoso virus se instala en su ordenador, espera unos días y, entonces, se activa en pantalla escribiendo *Free Kevin (Libera a Kevin)*, se refiere a Kevin Mitnick, probablemente el hacker más famoso de la década de los 90 que fue encarcelado por sus actividades). Este tipo de virus es molesto pero no causa ningún daño.

Sin embargo, se ha culpado a los hackers de otros sucesos como por ejemplo la repentina desaparición de millones de dólares de cuentas bancarias o la puesta en peligro de los astronautas del Shuttle. (La NASA niega que realmente se creara una situación de peligro cuando un hacker irrumpió en su sistema durante el aterrizaje de una nave). Todo el mundo recuerda los millones que se perdieron en productividad cuando se produjeron los ataques de los virus Melissa y ILOVEYOU. Los medios de comunicación recogen a diario sucesos relacionados con la seguridad.

El contenido de este libro

Este libro cubre todos los aspectos relacionados con la seguridad informática. Algunos de los temas recogidos incluyen:

1. Cómo mantener el anonimato al enviar correo electrónico, al inscribirse a un grupo de noticias o al chatear (co-

re electrónico, grupos de noticias y chats son totalmente opuestos a anónimo, aunque mucha gente, por error, se siente anónima cuando realiza estas actividades).

2. Bloquear la entrada de las "arañas" que pululan por la red intentando irrumpir en su ordenador mientras navega por Internet.
3. Evitar que otros le observen mientras navega y formen un perfil personal a partir de sus movimientos: qué sitios visita, qué lee, qué fotos mira, cuánto tiempo las mira, cuáles ignora, qué compra, cuándo compra y otros miles de datos. Al juntarlos, todos estos fragmentos de información le proporcionan a estos "extraños" (tanto si son particulares, empresas u organismos estatales) un retrato exacto y sorprendentemente detallado de su personalidad, estado de cuentas o información personal como su número de la Seguridad Social, etc.
4. Cómo las empresas se pueden defender de forma inteligente de los ataques de hackers, tanto si son ajenos a la empresa como si son empleados enfadados.
5. Cifrar su información de una forma sencilla y exhaustiva (de esta forma, si alguien consigue acceder a sus archivos o a su correo electrónico, no podrán sacar nada en limpio de los caracteres codificados).
6. Cómo evitar virus, tanto los ya existentes como los venideros.
7. Entender cómo el ordenador ha puesto el listón alto, tanto para la criptografía, que ha mejorado radicalmente, como para los intentos inevitables de los intrusos de descifrar los documentos cifrados.

Cowboys intelectuales

Este libro está dividido en tres secciones. En la primera parte, cuenta la intrigante historia de los cowboy intelectuales que recorren el campo electrónico, generalmente solos, buscando sistemas informáticos en los que irrumpir. Se hablará acerca de varios tipos de hacker: aquellos que sólo intentan demostrar la fragilidad de los sistemas de seguridad ("verdaderos" hacker), aquellos que quieren echarle un vistazo a la información de otras personas (whacker), aquellos que, sobrepasado el lado oscuro, intentan destrozarse los sistemas tras haber entrado en ellos y los novatillos llamados larvae. Verá cómo los hacker consiguen burlar los siste-

mas de seguridad de máquinas individuales y en red. Descubrirá dónde cuelgan e intercambian mensajes/notas en Internet (casi siempre es muy interesante escucharlos). Verá lo que puede hacer para proteger su ordenador personal o el de su empresa de estos visitantes indeseados.

Carnivore ataca

En la segunda parte, se trata fundamentalmente el cifrado y otras técnicas de ocultación de información que puede usar para proteger su privacidad. Podrá entender cómo funciona la criptografía y cómo usarla. Se hablará de técnicas relacionadas, como las firmas electrónicas y el *remailing* o reenvío, que protegen su información de un creciente intrusismo. Existe una amplia variedad de programas, algunos de los mejores gratuitos, que puede empezar a usar para ocultar la información de su disco duro o antes de enviarla a través de Internet. Otros programas bloquean el acceso de intrusos a su disco duro aún cuando tenga su ordenador conectado a la red continuamente.

Esta sección también trata varios temas relacionados con la seria amenaza que suponen los ordenadores para la libertad individual. Estos pueden grabar continuamente y sin coste alguno cada correo electrónico, cada compra, cada vez que presiona una tecla. Pongamos como ejemplo el sistema del FBI *Carnivore*.

Los mecanismos de *Carnivore* fueron instalados de forma secreta en los principales Proveedores de Servicios de Internet (ISP) en marzo de 2000, pero sólo se hizo público cuando a finales de julio EarthLink, uno de los mayores proveedores, rechazó instalarlo y lo denunció para evitarlo. Todo el tráfico de los proveedores fluye a través de una caja de *Carnivore*, no sólo la información relativa a delincuentes bajo investigación, sino todo el tráfico de Internet.

El FBI afirma que el *Carnivore* tiene la habilidad de distinguir entre el tráfico general (que puede ignorar) y las comunicaciones que puede interceptar legalmente. Insisten en que *Carnivore* sólo graba información relacionada con las investigaciones del FBI. Por supuesto, de esto podrían inferirse que utilizan información que lleva a nuevas investigaciones. Es como para sentarse a pensar.

Ahora multiplique *Carnivore* por varios cientos de "sabuesos" que andan pululando de aquí para allá desde su

teclado al sitio Web que está visitando. Se puede hacer una idea. No estamos hablando aquí sólo del FBI. Es probable que el FBI cumpla la ley y realmente ignore su pequeña y legal vida tal y como dicen. Esta es la última de nuestras preocupaciones. El problema es que hay cientos de sabuesos y nadie sabe cuántos o quién los utiliza.

¿Cree que estos sabuesos no se pueden permitir almacenar montones de datos acerca de usted o toda la actividad que tiene lugar en Internet? ¿Cree que si incluso pudieran almacenar todas las actividades de los ordenadores de todo el mundo no dispondrían de un mecanismo de búsqueda de datos interesantes? Reflexione de nuevo.

Como verá más adelante, los costes de almacenamiento de datos disminuyen a gran velocidad. Con los precios actuales, todo el correo electrónico que escriba a lo largo de su vida puede almacenarse por 20 pesetas y, probablemente, cueste 2 pesetas el próximo año o cuando los DVD regrabables sustituyan a los CD. La cuestión es que los ordenadores facilitan enormemente las tareas de recogida, almacenamiento y búsqueda de grandes cantidades de información.

Lleva menos de un segundo buscar palabras sospechosas, como *Bangkok*, en todo ese insignificante correo electrónico de 20 pesetas. Al acabar la búsqueda, un dispositivo marca todos los párrafos que haya leído o escrito que contengan la palabra *Bangkok*. Incluso no sería necesario leerlos, ya que en la parte superior aparece un análisis de los mismos. El ordenador le proporciona el porcentaje de veces que ha usado esa palabra a lo largo de su vida en comparación a la media; la frecuencia de palabras relacionadas como *Juárez*; su perfil financiero, legal y de desplazamiento en el contexto de ciertos tipos de ciudades extranjeras y sugerencia de castigos. (Sólo bromeo sobre los castigos... Espero.)

Dicho de otra forma, la STASI, la policía secreta de la antigua Alemania del Este, era muy constante y efectiva, pero los ordenadores son máquinas de una capacidad más elevada de observación y análisis de lo que jamás podría haber sido cualquier agencia de seguridad humana.

Deje de preocuparse por los virus

La tercera parte, tiene como objetivo la desmitificación de este tema casi siempre innecesariamente temido. "Un terrible virus golpea a empresas de todo el mundo. Una joven

niñera es detenida en la estación de autobuses. Se encuentran manzanas en mal estado en comedores escolares. ¿Funcionará alguna vez bien la Seguridad Social? Noticias a las 11".

Las noticias casi siempre exageran.

Hechos: es casi imposible que alguna vez tenga algún virus en su ordenador personal y esto se puede evitar tomando un par de medidas de precaución.

E, incluso, si le entrase uno, no causaría ningún desastre si hace regularmente copias de seguridad de sus archivos, proceso muy simple y económico. No hay razón alguna para no hacer copias en un par de minutos cada día. Si se acostumbra a hacerlo ni el peor virus del mundo podría hacerle nada.

Estoy de acuerdo en que los virus son unas criaturitas interesantes. *Mockingbirds*, muertos vivientes, bombas lógicas, gusanos, caballos de Troya, correos electrónicos agresivos, puertas traseras, zombis, el *Rat Dance*; todos componen un gran zoo de creaciones realmente inteligentes. Sin embargo, al igual que visita el zoo, disfrute con este tour de extraños animales pero no espere que le aparezca un pingüino en su cama cuando llegue a casa.

En la tercera parte de este libro podrá hacer una excursión por el variado mundo de los virus informáticos pero le sugiero que deje de preocuparse porque le entre un virus. A menos que sea el responsable de proteger el sistema informático en una empresa, no deje de dormir a causa de los virus. Simplemente recuerde hacer copias de seguridad. Si es el responsable de seguridad, este libro le enseñará cómo crear cortafuegos y otras defensas contra ataques.

¿Hay algún secreto en este libro?

Probablemente se estará preguntando si en este libro cuento detalles específicos sobre el hacking, exactamente cómo conseguir las contraseñas de cierto software, herramientas de hacker, números de la Seguridad Social de otras personas y otros trucos de hacker. Reflexioné un par de veces sobre esto pero no quería elaborar uno de esos libros super guay de publicación rápida que se centran en detalles concretos que no tienen nada que ver con la práctica habitual.

Sin embargo, al final decidí que debería dar algunos detalles. Asumo que la mayoría de los lectores que voy a tener

son buena gente y que, simplemente, intentan protegerse a sí mismos y a su empresa. Es más, los hackers conocen ya los trucos de su oficio. Aquellos que están empezando pueden obtener información fácilmente de otras muchas fuentes diferentes. Así que decidí dar pequeños detalles sobre los temas que aquí cubro.

Cómo contactar conmigo

Estoy a punto de cometer una estupidez pero allá voy. Me gustaría recibir sugerencias, comentarios, críticas o preguntas de mis lectores. Es verdad que dedico buena parte del libro a explicar cómo y por qué debería ocultar su dirección de correo electrónico a extraños.

Pero echemos pelillos a la mar porque mi interés en recibir información suya es mayor que mi miedo a los hacker y los que envían correo basura. Escribanme a earth@worldnet.att.net.

Peligro en Internet

Cuando se conecta a Internet, puede comunicarse potencialmente con cualquiera de los millones de usuarios que naveguen en ese momento. De repente, su pequeño disco duro pasa a ser un blanco perfecto.

Alguna gente, denominados hacker, whacker, cracker y otros apelativos inadecuados para ser reproducidos en este libro, se entretiene buscando estos discos duros y los utilizan. A veces, sólo figonean; otras, borran archivos o dejan virus, gusanos, bombas lógicas u otros elementos problemáticos.

Como verá en los próximos capítulos, todavía no conocemos la existencia de ningún hacker que se haya aprovechado de un usuario normal que tenga una conexión de alta velocidad permanente. Hay que aclarar que muchas empresas y particulares no informan de ataques de virus o de hackers del mismo modo que no se denuncian agresiones físicas. Muchas personas se avergüenzan y no quieren que su vulnerabilidad sea del dominio público.

Pero tampoco se puede saber qué deparará el futuro. Y los consejos destacados en este libro pueden ayudarle de una forma fácil y sencilla a proteger su sistema de intrusiones, ahora y en el futuro.

1.1. Como la miel a las abejas

No sea ingenuo cuando se conecte a Internet, especialmente si dispone de una conexión de banda ancha (de alta velocidad). Permitir compartir archivos e impresoras o puertos abiertos innecesariamente atraerá a los escáneres robotizados de los hackers como la miel a las abejas.

Pongamos por caso que se ha hecho con una conexión de esas nuevas líneas telefónicas de banda ancha (DSL) o un módem cable. ¡Guau! Las páginas de Internet aparecen rápidamente en pantalla y ya no tiene que esperar nada a que se descarguen las imágenes.

1.2. Sé dónde vives

Otra característica de la conexión de banda ancha es que no tiene que marcar ningún número de teléfono. Está permanentemente activada al igual que sucede con la televisión. Pero ahí está el problema: al estar siempre conectado, su puerta virtual está abierta a ese gran mundo malvado.

Las conexiones de banda ancha le proporcionan una dirección de Internet (IP) permanente. Esta dirección de Internet de su ordenador nunca varía. Párese un momento a pensar lo que esto significa: para un hacker esto es el equivalente de *¿sé dónde vives!*

Cuando usa la vieja y lenta conexión telefónica a Internet a través del módem, se le asigna una dirección IP diferente cada vez que se conecta. Al terminar la conexión, bien porque cierra el navegador, el programa de gestión de correo o apaga el ordenador, desaparece esa dirección IP temporal.

Pero con las conexiones modernas de alta velocidad, obtiene una conexión IP duradera y estable, como sucede con su número de teléfono o su domicilio. Más adelante se tratan con detalle los peligros a los que se enfrenta con estas nuevas conexiones, pero tiene que ser consciente de que se expone al ataque de los hackers al entrar en Internet con un ordenador que tiene una dirección IP permanente siempre activa.

Llamadas sin coste adicional a cualquier parte del mundo Vea cómo...

La dirección IP es un número único asignado a cada ordenador que se conecta a Internet. Por ejemplo, cuando pincha sobre un hipervínculo para visitar una página Web, las palabras que aparecen en el hipervínculo (como por ejemplo microsoft.com) se traducen automáticamente en una dirección IP. Las palabras comprensibles para el ser humano como microsoft.com se transforman en dígitos de una dirección IP comprensibles para el ordenador.

Las direcciones IP están formadas por cuatro números, separados por puntos. Aquí tiene un ejemplo típico:

212.53.166.236

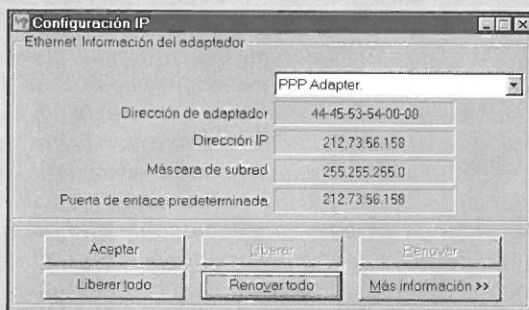
Puede ser que quiera darle su dirección IP a un amigo si tiene una permanente. Por ejemplo, se puede ahorrar un montón de dinero si su amigo está en otro continente. Puede utilizar la aplicación de Microsoft NetMeeting que le permite enviar y recibir mensajes de chat, enviar archivos o gráficos o, si tiene tarjeta de sonido, conectar un micrófono (sirve con una baratija de Radio Shack o de CompUSA) para mantener así conversaciones con un amigo que tenga el mismo equipo desde cualquier parte del mundo. Estas llamadas no tienen ningún coste adicional y puede hablar todo el tiempo que quiera. Hace poco estuve hablando durante una hora con un amigo que estaba en Atenas. Imagínese lo que habría costado si hubiese sido una llamada de teléfono convencional. Puede haber un poco de eco pero tampoco se puede decir que por teléfono se obtenga un sonido de alta fidelidad. El NetMeeting está dentro de las aplicaciones de Internet Explorer 5, programa que se adjunta con el Windows 98 y el 2000. Para activar el NetMeeting, pinche sobre el botón de Inicio, vaya a Programas y desplace la barra hasta que encuentre la carpeta. Si no lo encuentra de esta manera vaya a Inicio, Programas, Accesorios, Herramientas de Internet.

Si no lo ha usado antes, el propio programa le guiará durante el proceso de instalación. Una vez instalado, utilícelo



para llamar a su amigo de Atenas seleccionando Llamada, Nueva llamada desde el menú. Teclee la dirección IP de su amigo como se muestra en el siguiente gráfico.

Deben conocer para esto sus direcciones IP para teclearlas en NetMeeting y realizar así la conexión. Para saber cuál es su dirección IP, conéctese a Internet a través de un navegador o de un programa de gestión de correo electrónico. Una vez dentro, pinche sobre el botón de Inicio de Windows, elija Ejecutar y teclee WINIPCFG. Acepte y la utilidad IP de Windows se ejecuta como se muestra en el siguiente gráfico:



1.3. Explorar los protocolos de Windows

Windows incluye tres protocolos primarios (conjunto de reglas) que facilitan el intercambio de comunicación entre ordenadores. Estos tres protocolos son: IPX/SPX, NetBEUI y TCP/IP.

- **IPX/SPX:** Conjunto de dos protocolos que permiten las interconexiones en red a gente que usa servidores y clientes NetWare de Novell.
- **NetBEUI (NetBIOS Extended User Interface):** Extensión de NetBIOS, utilidad que facilita las comunicaciones LAN. NetBEUI fue creado originalmente por IBM, pero ha sido tomada por Microsoft para NT y Windows 95/98.

- **TCP/IP:** Este común estándar de Internet puede usarse en redes LAN y WAN y es el protocolo de comunicación en las conexiones a Internet. La parte TCP (*Transmission Control Protocol*, Protocolo de control de transmisiones) divide su mensaje en pequeñas partes (paquetes) y lo recompone cuando el programa TCP del ordenador receptor los recibe. La parte IP (*Internet Protocol*, Protocolo de Internet) se ocupa del direccionamiento, es decir, se asegura de que cada uno de los paquetes está siendo enviado al ordenador correcto (tenga en cuenta que hay muchos ordenadores conectados a Internet a cualquier hora del día). Junto con el TCP/IP funcionan otros protocolos como el **FTP** (*File Transfer Protocol*, Protocolo de transferencia de archivos) y **HTTP** (*Hypertext Transfer Protocol*, Protocolo de transferencia de hipertexto). El hipertexto es el lenguaje informático que se utiliza para describir la mayoría de los elementos de una página web, como colores, tipos de fuentes, etc.

El objetivo de todos estos tipos de protocolos es que se pueden conectar entre aplicaciones, como el navegador y el hardware que tiene ese cable que se desliza hasta la conexión telefónica o de cable. Este cable es su conexión física a Internet y, precisamente a través de él, puede entrar el hacker en su ordenador. Sin tener que llegar a cortarlo o a cortar el suministro eléctrico puede protegerse del hacker especificando, por ejemplo, que no permite compartir archivos. De esta forma, aunque tenga puertos abiertos en su ordenador, la mayoría de los hackers no podrán acceder a su disco duro. (En cualquier caso, no debería tener ningún puerto abierto, como veremos más adelante.)

1.4. La seguridad en Internet con Windows

Al instalar cualquiera de las versiones de Windows, no ofrecen, por defecto, la mayor de las protecciones contra los hackers de Internet. Windows está diseñado para permitir varias clases de trabajo en red, es decir, conectar varios ordenadores entre sí, bien de forma local (LAN, *Local Area Networking*, Conexión en red de área local) bien a gran dis-

tancia (WAN, *Wide Area Networking*, Conexión en red de área amplia.)

Obviamente es un intercambio entre compartir y proteger. Por un lado, quiere tener la posibilidad de compartir archivos y otros recursos, como impresoras, con sus amigos y compañeros de trabajo. Por el otro, no quiere compartir cosas con extraños, especialmente hacker.

En capítulos posteriores entraremos con más detalle en alguno de estos aspectos, pero está bien tener una visión general del problema en sí.

1.4.1. Evitemos compartir archivos

Para la mayoría de los sistemas es tan perjudicial dejar activada la opción de archivo compartido que no me cansaré de repetir este consejo básico a lo largo de todo el libro. Desactívela en Inicio, Configuración, Panel de Control. Haga doble clic en el icono de Red. En la pestaña de Configuración del cuadro de diálogo, haga clic en el botón **compartir archivos e impresoras**. Desactive el cuadro situado antes de **permitir que otros usuarios tengan acceso a mis archivos**. Pinche sobre **Aceptar** dos veces para cerrar los cuadros de diálogo.

La mayoría de los usuarios no tenemos razón alguna para permitir que IPX/SPX o NetBEUI estén activamente disponibles en nuestra conexión a Internet. Ni tampoco que las características de nuestro servidor personal de Windows anden por Internet.

El protocolo TCP/IP es el que casi todos los usuarios necesitamos para navegar por la red y enviar correo electrónico. En términos prácticos, no tendría necesidad de tener abiertos puertos (entradas), exponerse a ser tomado como servidor (como Servidor Personal de Páginas Web) y otros elementos de su ordenador.

X 1.5. Llamar a su propia puerta

Basta ya de teoría. Intentemos hacer ahora un experimento para ver a qué nivel queda su ordenador expuesto a Internet. Un sitio web fantástico es Gibson Research Corporation que tiene varias aplicaciones útiles, entre ellas **Shields Up!**

Por ahora, puede usar estas utilidades de test para probar la vulnerabilidad de su sistema. Vaya al sitio: www.grc.com.

1.5.1. Pruebe sus escudos y puertos

Haga clic sobre los vínculos de *Shields Up!* (tiene que hacer clic en dos vínculos de dos páginas diferentes) para ir a la página del test. Baje la barra de desplazamiento y verá dos opciones: *Test My Shields!* y *Probe My Ports!*

Nota: Si trabaja en red, deberá obtener un permiso del departamento informático antes de llevar a cabo este tipo de test.

Si no trabaja en red, o ha obtenido dicho permiso, haga clic sobre el botón *Test My Shields!*. Identificará su dirección IP y comenzarán los tests de prueba. En mi ordenador se detectaron una serie de problemas (explicados en inglés, eso sí, no técnico). El puerto 139 estaba abierto a todo aquel que quisiera verlo y entrar. Este puerto le permitió a *Shields Up!* conectarse al puerto de mi archivo de NetBIOS y a compartir impresora. Dicho claramente, mi ordenador tenía un puerto abierto que podría haber sido explotado por cualquiera que lo hubiese detectado. Recuerde que los hackers utilizan programas que merodean por la red probando las direcciones IP para encontrar entradas de este tipo y acceder al ordenador de cualquiera. Yo no quiero que haya extraños que usen esta entrada para conseguir acceder a mi disco duro y obtener información sobre mí o mi disco duro y esto es justo lo que está pasando. ¡Tengo que tomar medidas para defender o para cerrar el puerto 139!

1.5.2. Es horrible que se difunda su información personal

Otras pruebas llevadas a cabo con *Shields Up!* durante el proceso de comprobación de defensas me demostró que, a pesar de tener bloqueada la conexión vía NetBIOS, mi nombre de usuario, mi nombre de máquina y mi grupo de trabajo se difundían por Internet. No tengo activadas las opciones de compartir archivos e impresora, por lo que no se permite ninguna conexión a través de este puerto abierto NetBIOS. Pero estarán de acuerdo en que es muy desagradable que parte de mi información personal ande pululando por ahí para que la vea cualquiera.

Otras pruebas me revelaron que mi dirección de MAC (*Media Access Control*, Control de Acceso de Media) de mi tarjeta Ethernet (conexión en red) podía ser leída por cualquiera. Con esta tarjeta conecto mi ordenador al módem por cable. Cualquier extraño podría haber visto no sólo mis nombres de usuario, equipo y grupo de trabajo, sino también identificar mi ordenador. El MAC es el único número de serie de Ethernet. Ninguna otra tarjeta tiene ese número. Al igual que el ADN, es exclusivamente suyo. Queremos que ciertas actividades en Internet sean totalmente privadas, como transacciones bancarias o inversiones. Hay un único ID asociado con mis movimientos en Internet, y si este es público, me preocupa mucho.

Cuando activé el botón de **Shields Up! Probe My Ports**, los resultados confirmaron mis problemas. Comprobó 10 puertos normales (recuerde que hay más de 60.000 puertos, así que seguro que esta versión de prueba sólo analizó aquellos más buscados por los hacker). De nuevo, se demostró que el puerto 139 estaba abierto. Steve Gibson, el creador de la utilidad Shields Up!, dice en la misma página en la que le salen los resultados que "El archivo de puerto compartido de NetBIOS es el agujero de seguridad más grande en los equipos conectados en red con Windows".

Fíjese que yo tengo una red abierta, aunque yo no trabajo en red. Es posible estar expuesto a conexiones en red dentro de Internet (que en sí misma es una enorme WAN), aún cuando, al igual que yo, no pertenezca a ninguna red física. Trabajo en casa, solo y utilizo un único ordenador que no está conectado a ningún otro. ¡Ni tan siquiera tengo vecinos cerca! Me creía a salvo, que era anónimo y que tenía privacidad y seguridad al conectarme a Internet. En mi caso, mi tarjeta Ethernet, requerida por la empresa que me proporciona la conexión por cable, me identifica ante el resto del mundo. Además, la comprobación de los puertos demostró que los 10 puertos analizados eran reconocidos por mi ordenador. En otras palabras, le estaba diciendo a extraños "este puerto existe en este ordenador que tiene esta dirección IP, aunque ahora mismo está cerrado".

Lo ideal sería que todos los puertos fuesen invisibles a las comprobaciones externas. Estas nunca serían capaces de detectar la existencia de ningún puerto y mucho menos si están abiertos o cerrados.

Afortunadamente, hay soluciones para todos estos problemas.

1.5.3. Las mejores soluciones para las intentonas de los hackers

Si, al igual que yo, obtiene resultados preocupantes con los tests de Shields Up!, puede leer las distintas páginas en el sitio Web de Steve Gibson. Allí le explican cómo bloquear intentos de entrada, cerrar puertos y otras soluciones inteligentes. Otra forma es instalar la excelente y potente utilidad gratuita ZoneAlarm. Este cortafuegos personal cubrirá su ordenador con una defensa que lo hará menos tangible que la sonrisa de un fantasma. Si quiere instalarlo ya, váyase a las instrucciones recogidas en la sección "Instalación de ZoneAlarm" más adelante en este libro. Es fácil, potente y rápido. Y, además, gratuito, a menos que sea una empresa o una institución administrativa o educativa.

Phone Phreaks

Los Phone Phreaks (entusiasta del teléfono) son los antepasados directos de los hacker actuales. Mediante el uso de *war dialers*, "hurgar en la basura", ingeniería social y otros procedimientos, estos primeros hacker crearon métodos y técnicas que se utilizan todavía en la actualidad para romper la seguridad de organizaciones pequeñas y grandes. No intentaban romper la seguridad de redes informáticas, sino que su reto era entrar en los sistemas de las compañías telefónicas y escuchar llamadas ajenas, realizar llamadas gratuitas, enviar facturas enormes a sus enemigos y otras maneras de moverse por la infraestructura de la vieja Bell sin que los pillaran.

Un *war dialer* (generador de tonos) es un programa que marca continuamente un conjunto de números de teléfono buscando aquellos que responden con una señal electrónica que no sea la voz. Algunos de estos programas son capaces de diferenciar entre fax, módem u otras clases de comunicaciones electrónicas, por ejemplo, la respuesta de un sistema informático activo. Con las actuales conexiones permanentes a Internet (DSL o módem de cable) los *war dialers* pueden utilizarse para entrar en cualquier conexión activa. Un *war dialer* no es lo mismo que un *daemon dialer* (demonio), pues este último llama continuamente al mismo número. El *daemon dialer* se puede utilizar para acceder a un número con gran demanda de llamadas o para bloquear un sitio Web u otro tipo de conexión saturándolos. Esta acción de llamar a un mismo número constantemente ralentiza o detiene el sistema y se denomina ataque de denegación de servicio.

El *hurgar en la basura* es una técnica por la que el *phreak* obtiene información revolviendo en la basura, como manua-

les obsoletos o algún hardware viejo pero todavía útil que alguien ha tirado. Un famoso ejemplo son los primeros *phreaks* que iban a hurgar en los contenedores de basura situados detrás de los edificios de la compañía telefónica Southern Bell y conseguían así listados de contraseñas, sistemas de enrutado y otra información técnica muy útil.

La ingeniería social se refiere a aquellas irrupciones en el sistema de seguridad basadas en el engaño y la mentira a personas y no en técnicas propias de un hacker a través de software o hardware. Entre las técnicas de ingeniería social figuran hacerse pasar por un superior de las oficinas centrales, por alguien del FBI, un técnico informático con un problema urgente, etc. Muchas veces, la ingeniería social es la forma más efectiva para burlar un sistema de seguridad. Puede poner un ordenador en una habitación sellada con paredes de tres metros de ancho, pero si un empleado que conoce la clave de entrada es un cotilla, se siente solo o es vulnerable en cualquier otro sentido, ni siquiera muros de 15 metros protegerían el sistema. La seguridad se compone de una cadena de elementos relacionados: cortafuegos, contraseñas, *shredders*, sistemas de alarma, salas de seguridad, etc. Pero, como reza el viejo dicho: la fuerza de una cadena de la seguridad viene dada por su eslabón más débil. Y, con frecuencia, el eslabón más débil es una persona.

2.1. ¿Quiénes son los *Phone Phreaks*?

Se autodenominaron *Phone Phreaks*. Se asombraría de la cantidad de términos inteligentes creados por estos rebeldes intelectuales y fuera de la ley que se mueven en los límites de la información. Los hacker, cracker, whacker, da igual como los llame, suelen ser mañosos y, a veces, originales.

Uno de mis favoritos de toda esta avalancha de neologismos es la palabra *warez*. Warez se refiere al software comercial que ha sido craqueado (se ha eliminado la contraseña o cualquier otro sistema de protección anticopia) y el warez puede circular y ser usado libremente por cualquiera. No es ético utilizar estas copias ilegales de software con licencia. Reducen los ingresos legítimos y justos de programadores y demás personal relacionado con la producción de software profesional. Una vez dicho esto, ha de reconocer que el que creó esa palabra fue realmente ingenioso (como en la música hip-hop, en la jerga hacker gustan de usar la letra z).

Phreaking significa intentar craquear el sistema telefónico. El objetivo primario es evitar pagar las llamadas realizadas. En un principio (a finales de los 70 y principios de los 80) los *phreaks* utilizaban su destreza técnica para imitar los pitidos y sonidos electrónicos que activaban y manipulaban los circuitos telefónicos. La compañía telefónica contraatacó con componentes electrónicos más difíciles de craquear. Desde finales de los 80, el *phreaking* descendió por la escalera de la moralidad desde sus retos intelectuales originales a, simplemente, infringir la ley, como por ejemplo, robando números de tarjetas telefónicas.

Tiene cierta lógica atacar sistemas telefónicos o de buzón de voz si se quiere aprender a craquear sistemas informáticos. Los sistemas telefónicos son sistemas informáticos primitivos y, normalmente, con un débil sistema de seguridad. Entrar en ellos es un buen entrenamiento. Es más, entender los sistemas telefónicos es una buena base para encontrar diferentes maneras de entrar en una red de área local.

Proteja su empresa

Si dirige una empresa, haga que sus contables busquen regularmente en sus facturas telefónicas cualquiera de las siguientes irregularidades. Si encuentra algún problema, contacte con la compañía telefónica inmediatamente. Aunque este no sea su caso, también debería echar un vistazo a su factura telefónica. Fíjese en los siguientes aspectos:

- *Llamadas internacionales o intercontinentales desconocidas.*
- *Un incremento repentino de llamadas.*
- *Cualquier cambio en las rutinas, sobre todo el aumento de llamadas fuera de horario laboral, como noches o fines de semana.*
- *Una subida repentina del tráfico entrante, especialmente de gente que cuelga u otro tipo de llamadas extrañas.*
- *Llamadas a números 900, lo que podría ser también un problema interno.*
- *Cambios en la actividad de su tarjeta de crédito.*

2.1.1. Sintetizadores malvados

Si es capaz de imitar los sonidos del sistema internacional de tonos digitales, puede usar los tonos especiales lla-

mados "C5" para burlar el sistema de seguridad normal de la compañía telefónica. Los tonos C5 no están disponibles en teléfonos normales por lo que no se pueden generar presionando las teclas. Sin embargo, con estos tonos puede forzar el paso de estas llamadas a través del sistema internacional de telefonía de manera gratuita.

No hay un gran paso entre construir este tipo de supergenerador de tonos y entre vender estos aparatos y programarlos para permitir llamadas gratuitas a otros países. A la gente que le coge el gusto a hacer esta trampa, le resulta fácil pasar a otras clases de raterías relacionadas: copiar tarjetas magnéticas (por ejemplo, tarjetas de crédito o telefónicas); reprogramar o reemplazar chips en los teléfonos de los coches o desbloquear los receptores de la televisión por satélite para conseguir ver gratis todo lo que quieran.

También se sabe que los *phone phreaks* han intervenido llamadas telefónicas ajenas, han manipulado las facturas de sus enemigos para que fueran elevadas, han amañado llamadas internacionales gratuitas, han desconectado la línea de aquellos que le eran antipáticos, han fisgoneado en notas universitarias y en bases de datos de exámenes y han tomado parte en actividades que se mueven entre travesuras y fraude directo y robo. Aunque las compañías telefónicas intentan ayudarle cuando alguien le roba su número de tarjeta telefónica, el propietario es el responsable último de las llamadas realizadas con su número de tarjeta.

***Nota:** ¿Sabía que la compañía telefónica puede escuchar cualquier conversación? Yo no, hasta que averigüé que, con la combinación adecuada de pitidos, un phone phreak puede acceder a todas sus llamadas privadas.*

2.1.2. Bip, bip

Los teléfonos celulares son increíblemente inseguros. Por unas 25.000 ptas. puede comprar un radio escáner, capaz de interceptar llamadas de teléfonos celulares. En estas llamadas se intercambia toda clase de información privada. Se supone que no se pueden escuchar llamadas ajenas; de lo contrario, estaría infringiendo la ley, concretamente en Estados Unidos la Ley de la Privacidad en las Comunicaciones Electrónicas.

Por 22.000 ptas. puede grabar y descodificar los pitidos cuando la gente utiliza el teléfono convencional. Si usa un aparato llamado *grabador de tonos*, puede conseguir números de tarjetas de crédito y de tarjetas telefónicas. Cada teléfono celular tiene un número único que lo identifica y por el que se sabe a quién hay que facturar. El Número de Indetificación del Móvil y el Número de Serie Electrónico (MIN y ESN) pueden capturarse cuando se transmiten al iniciar una llamada y ser programados en otro teléfono. El coste medio de un teléfono programado en el mercado negro ronda las 70.000 ptas. al mes y dispone de un número ilimitado de llamadas a cualquier lugar. Para estafadores y traficantes, esto puede ser una ganga. La industria de la telefonía celular está luchando contra esto, especialmente cifrando las señales digitales.

Hackers, crackers y whackers

Suele ser un acto de rebeldía. La mayoría de los programadores de virus y los hacker se sienten al margen de la sociedad. Son personas con un nivel de inteligencia relativamente alto pero no quieren un trabajo convencional. Unas veces quieren ser famosos, o que se les reconozca su inteligencia, pero no suele ser el dinero su primer objetivo. Se quejan de que la información debería ser libre, pero nadie da nada a cambio de nada. Si alguien produce información útil espera, con todo su derecho, ser recompensado para poder pagarse su manutención y vivienda. La teoría de los hacker de la "libre información" ignora también la cantidad de trabajo que precedió a su acceso a Internet y a los ordenadores.

Con frecuencia, un hacker es la versión intelectual de los adolescentes que se dedican a hacer pintadas. Sintiendo vacíos de poder/impotentes, al menos pueden irrumpir en el ordenador de otra persona, dejar su huella y, de paso, demostrarnos a los demás que existen. Los perros marcan su territorio y así se comportan también personas con problemas de integración social. No reciben la suficiente atención en su círculo social, así que quieren demostrar hasta dónde pueden llegar.

Por supuesto, es imposible describir a tan vasto grupo de gente con un único perfil psicológico. Es cierto que no todos los hacker están marginados socialmente ni son todos unos desagradables. Es bien seguro que algunos son guapos, populares y fibrosos. Uno o dos.

Cuando finalmente maduran, los más cualificados suelen pasarse al otro bando. Es la historia de siempre: unos protestan y se quejan de su situación, se les prometen maravillas y acaban trabajando para el gobierno o para cualquier empresa que en su día criticaron.

Es probable que Kevin Mitenik, el hacker más famoso, testificase recientemente en el Congreso. Muchos hacker han sido contratados por grandes empresas como especialistas en seguridad. Otros se han unido para crear empresas consultoras de seguridad.

A menudo oírás decir que hay que distinguir entre aquellos que sólo curiosoan (los verdaderos hacker) y los que son destructivos (sombremos negros, whacker y cracker). La distinción tiene un algo de autodefensa, pero algo hay de verdad en todo ello. Dicen los puristas que un verdadero hacker sólo está interesado en el desciframiento, ver si puede romper el sistema de seguridad o aprender cómo funcionan redes y sistemas. Su interés es meramente académico y nunca causan daño. Son como los ornitólogos salen para observar a los pájaros, sin causarles daño alguno porque son el objeto de su fascinación.

Esta definición se adecua a muchas personas. De hecho, es una descripción bastante ajustada de cualquiera que disfrute aprendiendo. Pero los medios de comunicación y el público general ignoran la distinción. Y, siendo sinceros, muchos hacker se mueven entre las actividades aceptables y las inaceptables. Cuando se les ofreció dinero, los *phone phreaks* se fueron a trabajar a empresas de telefonía. ¿Su trabajo? Seguridad, evidentemente. Han enseñado a las compañías telefónicas cómo prevenir futuros ataques de los *phreaks* y cómo reforzar sus sistemas para evitar problemas.

De la misma forma, podemos encontrar muchos casos de hacker que empezaron como tales para acabar trabajando para el bando contrario como consultores de seguridad. Puede que no hayan llegado al punto de llevar esas tiesas camisas blancas y corbata, pero desde luego han renunciado a lo que eran y han preferido tener un bonito coche y una seguridad económica. Es la vieja historia del rebelde que asienta la cabeza.

3.1. Los hacker novatos en la rampa de salida

Los hacker tradicionales están consternados ante el creciente número de adolescentes ignorantes y cabreados que usan el software de otras personas para hacer daño. Estos chavales apenas tienen conocimientos de programación. No podrían entrar ni en un sistema con un pésimo sistema de

seguridad. Sin embargo, pueden hacerse fácilmente con scripts que pueden llegar a hacer mucho daño, ya que estos programas están en cualquier motor de búsqueda, preparados para ser descargados.

Estos programas pirateados, listos para ser ejecutados, pueden ser usados por cualquiera que desee atacar otros ordenadores, sin que sea necesario ningún tipo de conocimiento. Estos novatos causan caos en los sitios Web de los demás. Lanzan ataques de *muertos vivientes*, zombis, denegación de servicio inundando dichos sitios con intentos de conexión en masa. Aún así, no saben lo que hacen. Incluso se realizan convenciones muy populares que, en la actualidad, atraen a más novatos que a los hacker reconocidos. La conferencia DEF CON 00 que se celebró del 28 al 30 de julio del 2000 en Las Vegas estaba tan llena de chavales, que más bien parecía una convención de Star Trek.

3.2. Cómo hablarle a un whacker de un hacker

Existen subcategorías dentro de la comunidad hacker, definidas por la forma en la que aplican la ética informal del hacker. Los hacker (incluso los principiantes, denominados en jerga, *larvae* o etapa de gestación) intentan explorar y entrar en otros sistemas operativos y romper códigos supuestamente de seguridad, pero no pretender causar daño, ni robar dinero ni información. El objetivo principal de estos hacker es intentar asegurar la libertad de información creando accesos a los ordenadores y que la información contenida en ellos no tenga barreras. A estos hacker también se les conoce con el nombre de samurais.

Por supuesto, siempre está nuestra privacidad en peligro, y en último término, también nuestras libertades individuales, debido a la creciente acumulación de nuestros datos personales por parte del gobierno. La comunidad hacker tiene parte de razón cuando afirma que, al irrumpir en los sistemas y observar los contenidos de las enormes bases de datos de gobierno y empresas, proporcionan un equilibrio de poder en la era de la información.

Los hacker también destacan otras virtudes morales y beneficios prácticos de su actividad. Al frenar la velocidad de funcionamiento de sitios de comercio electrónico, mediante

ataques de denegación de servicio en los que de forma continua y constante sobrecargan el servicio de recogida de llamadas, fuerzan a que esos sitios mejoren sus sistemas de seguridad ante tales peligros. Al romper redes supuestamente seguras, obligan a que se refuerce ese sistema. Claro está que este último razonamiento también podría aplicarse cuando se realizan malos actos, por lo que suena un poco a falso. La principal motivación de un hacker es la curiosidad, mezclada quizás con un deseo de reconocimiento. La afirmación de que sus esfuerzos funcionan como llamadas de aviso suena vacía, *a posteriori* y, como mucho, terciaria.

Sin embargo, deberíamos intentar seguir la lógica de la comunidad de los hacker. Es muy útil entender cómo se ven a sí mismos, lo cual se puede deducir de sus afirmaciones y de las distinciones que quieren establecer entre ellos mismos.

Los whacker se definen como posibles hacker cuya actividad está confinada a la investigación de sistemas sin intentar crear grandes rupturas en el sistema de seguridad. Los cracker son hacker que se han pasado de la raya y están interesados en el robo de información, en hacer daño (borrar el disco duro) y acabar, a veces, con sistemas enteros.

Los hacker ven con muy malos ojos a los cracker ya que son muy poco sofisticados y sólo pretenden hacer daño. Aportan mala fama a la comunidad. Si bien los hay muy inteligentes, la mayoría son simples e infantiles. Los cracker sustituyen la insistencia de la "fuerza bruta" y un pequeño grupo de trucos repétidos para aprovecharse de famosos talones de Aquiles de sistemas de seguridad por una verdadera inventiva y una técnica sofisticada. Otros utilizan programas ya escritos que descargan y difunden pero que no entienden.

La mayoría de los cracker no están muy cualificados; simplemente son muy pacientes y carecen de ética. Los cracker se asocian entre sí en pequeños grupos encubiertos, diferentes de la comunidad abierta para intercambio inteligente de información que forman los hacker. Sin embargo, hay que percatarse que la diferencia entre los cracker y los hacker no existe en los medios de comunicación ni para el público general. Y, por supuesto, los cracker se autodenominan hacker.

En un análisis final, sin embargo, el comportamiento de un hacker es el equivalente a entrar en una casa y merodear por ella para ver qué es lo que hay. Aunque no es tan grave como robar o prenderle fuego, el simple hecho de haber entrado es, para la mayoría de los humanos, algo que no se puede hacer.

3.3. Los hacker con virus

Una de las primeras cosas necesarias para cualquier hacker es entrar en el sistema de la víctima. Generalmente, esto implica conseguir una contraseña, ya que la mayoría de redes informáticas y sitios Web exigen una clave para permitir el acceso. En capítulos posteriores, explicaré algunos detalles de cómo se puede burlar un sistema, pero analicemos ahora el problema de los virus que almacenan contraseñas.

Una versión de un ataque por la fuerza bruta es mandar un virus a una red e intentar conseguir la clave de acceso. Para entrar en la mayoría de los sistemas se necesitan dos elementos: una contraseña y un nombre de usuario. Hay virus que pueden almacenarlos.

Un hacker puede enviar un virus a un sistema que, posteriormente, se incorpora él mismo al procedimiento de entrada al sistema. Pero ahora viene el problema: ¿Cómo le llega al hacker la información? Evidentemente, utilizar una cuenta de correo electrónico sería una locura, ya que sería muy fácil de localizar. Nadie programa un virus e incluye luego su cuenta de correo electrónico. No, a menos que haya perdido la cabeza.

¿Qué es lo que debe hacer un hacker rabioso? ¿Cómo puede recibir la información? Existen formas de crear direcciones de correo electrónico que no revelen su identidad.

También se puede "robar" la cuenta de correo electrónico de otra persona, y, a continuación, enviar y recibir mensajes de correo redireccionándolos a localizaciones donde puedan ser interceptados. Algunas personas han empezado a extender un virus enviándolo desde una cuenta de correo robada.

3.4. Cómo enviar correo electrónico o mensajes a grupos de noticias de forma anónima

No es mi intención en este libro ofrecer a los posibles hacker (o whacker o cracker) recetas específicas que les enseñen, paso a paso, cómo robar información, romper el sistema de seguridad o causar daño de alguna otra forma. Por ello describo diversas técnicas de forma genérica y no doy los detalles específicos del proceso en sí mismo.

Sin embargo, en algunos casos, tenemos todos una razón de peso para utilizar técnicas de hacker y, en esos casos, sí que doy detalles específicos. Más adelante se habla de un programa que puede cifrar su información de una forma tan efectiva que, en mi opinión, nadie podría craquearla. Está en su pleno derecho de proteger sus datos lo máximo que pueda.

También hay razones válidas, en mi opinión, para aprender a enviar correos electrónicos que no dejen rastro, de la misma forma que creo que puede haber un propósito social legítimo en el envío anónimo de cartas. En ciertas ocasiones, puede que quiera expresar su parecer a la administración, al supermercado de al lado, al jefe o cualquier otro sin querer que se conozca su identidad. Es decir, si quiere enviar correo electrónico o dejar mensaje en un grupo noticias que no le identifique, intente usar el programa *Ghost Mail* que se puede descargar desde varios sitios (para buscarlo, utilice *Google*, *Yahoo* o cualquier otro motor de búsqueda, o pruebe en esta dirección: <http://download.mycomputer.com/detail/60/14.html>).

Otra forma de enviar información de forma privada sería codificar las distintas contraseñas y nombres de usuario antes de reenviársela a uno mismo. Sin embargo, el sistema de codificación que se use debe residir dentro del virus porque el virus es el agente secreto que trabaja dentro de otro ordenador. Introducir ese sistema dentro del virus implica que alguien lo puede descifrar cuando lo desensamble (lea las instrucciones).

Otra táctica es enviar la lista de contraseñas a un grupo de noticias no supervisado (algunos grupos de noticias de Internet están moderados, de forma que nunca se les envían mensajes extraños, pero la mayoría no se supervisan y los llenan de correo basura). Por supuesto, el hacker podría querer cifrar las contraseñas y nombres de usuario capturados antes de enviarlas a un grupo de noticias.

3.5. Hablando de correo basura: cómo deshacernos de él

El correo electrónico basura no es un virus, pero puede ser muy irritante. No va a dañar su disco duro, pero satura su bandeja de entrada de cosas que no quiere leer. Es probable que le llegue este tipo de correo y no que le entre algún virus o le ataque un hacker. ¡Danesas calientes! ¡Engañe a su proveedor de TV digital!

Veamos a continuación cómo enfrentarse a la amenaza contra su placentera actividad informática: los temidos envíos de correo basura.

Mucha gente cree que si compra en sitios de comercio electrónico, como Amazon, recibirán mucho correo basura. Falso. Ninguna empresa de comercio electrónico respetable arriesgará su reputación vendiendo los datos de sus clientes. Si eso se supiese (y se sabría), se estarían perjudicando gravemente.

Otras personas temen que su Proveedor de Servicio de Internet (ISP) les proporcione los números de cuenta a la gente que se dedica a hacer estos envíos. Falso. Los proveedores intentan por todos los medios filtrar el correo basura.

El correo basura perjudica seriamente la eficiencia de Internet. Los expertos aseguran que un 10% de todo el tráfico de correo electrónico es basura. Reflexione un momento: su proveedor de Internet tiene que añadir un servidor extra para el correo basura por cada nueve mensajes normales de correo. ¿Y quién paga por ese hardware extra? Nosotros: Usted y yo.

Probablemente haya oído hablar de *crawlers*, *bots*, *spiders* y otras criaturas que, incansablemente, navegan por la red buscando direcciones de correo electrónico para enviárselas a sus creadores. Los amos de estas repugnantes criaturas, esas personas que envían millones de mensajes al día, esperando que le responda un 0,01% mientras saturan egoístamente Internet y la ralentizan.

Este porcentaje es elevado. Imagínese que cada día cien personas le envían 5.000 pesetas.

Seguro que se está preguntando cómo hay alguien que puede responder a esos mensajes absurdos e, incluso, escritos con faltas de ortografía. ¡No trabaje más! ¡Le garantizamos un plan único para usted! (Generalmente, ni siquiera se molestan en usar un corrector ortográfico).

Cuando trabajaba para una revista, el director de distribución no explicó en qué consistía el envío masivo de correo electrónico. Muchos están solos, otros se sienten confusos, quieren hacer algo útil y otros ya han perdido el norte. El envío masivo de correo electrónico proviene en su mayor parte de toda esta gente insatisfecha.

Pero merece la pena. Solos, confusos, locos o masoquistas tiene también tarjetas Visa activas.

La mayoría de nosotros entendemos lo que hay detrás de mensajes del estilo:

¡¡¡¡Richard Mansfield ha ganado 10.000.000.000.000 de pesetas!!!!

(...es lo que pondrá en el cheque si gana nuestro concurso).

Pero otras pobres criaturas, alrededor del 0,01 por ciento no lo entienden y envían dinero con la esperanza de que sus sueños se hagan realidad. Muchos organismos estatales de Estados Unidos han emprendido acciones legales contra este tipo de actividad cuando se produce a través de correo basura. Pero Internet no está protegida hasta ese punto.

A continuación explicaremos algunos pasos que puede dar para evitar el correo basura

3.5.1. Omita la dirección de correo electrónico

Si tiene una página Web, no incluya en ella su dirección de correo electrónico. Cuando instaló Windows por primera vez es probable que haya usado el asistente para la conexión a Internet que le pidió su dirección de correo de noticias. Y, con lo confiado que es, tecleó su verdadera dirección.

Reflexionemos sobre esto. Es cierto, al incluir su dirección de correo electrónico, abre la posibilidad de que otro miembro del grupo de noticias pueda contestar sus mensajes vía correo electrónico, en vez de colgarlos en el grupo.

Al escribir su verdadera dirección en grupos de noticias abiertos también atrae a esas arañas como si fuera una mosca volando alrededor de su tela. Como resultado, es posible que comience a recibir más correo basura que antes.

Le insto a que utilice una dirección falsa en este caso. Llegar con un nombre raro como Z. Z o Zorro o cualquier otro nombre popular confunde a las arañas y también avisa a otros miembros del grupo de noticias de que no intenten enviarle correo. Siempre pueden colgar una respuesta en el grupo. Pero su nombre falso debe adecuarse a la puntuación de Internet. No puede escribir simplemente Zorro, sino algo como `zorro@cualquiergrupo.net`.

De forma alternativa, si quiere permitir que los miembros del grupo de noticias le envíen correo puede usar un truco. Cambie su dirección de la forma que describiremos en el siguiente apartado pero no la encubra totalmente. Por ejemplo, si su verdadera dirección es `john12@aol.com`, teclee algo por el medio, como por ejemplo, `john12hotdog@aol.com`. Después, al final de cada mensaje que envíe al grupo, escriba esta frase para indicar a los otros miembros como pueden contactar con usted:

Si me quiere responder, elimine hotdog.

3.5.2. Disfrace su dirección de correo

Siga los siguientes pasos para cambiar su correo electrónico dentro de su grupo de noticias:

1. Elija Herramientas, Cuentas cuando esté dentro del grupo de noticias en Outlook Express. Aparecerá el cuadro de diálogo de las cuentas de Internet.
2. Haga clic sobre su grupo de noticias activo, que probablemente esté ya seleccionado cuando entre.
3. Haga clic en el botón de **Propiedades**.
4. Cambie su dirección real por algo parecido como SeñoraPeluquería@lacanet.es.
5. Haga clic sobre **Aceptar** y se cerrará la hoja de **Propiedades**.
6. Cierre.

3.5.3. Filtrar

Los navegadores de Internet le permiten especificar qué mensajes entrantes quiere rechazar. Puede describir una serie de condiciones que detendrán el acceso de cierto correo electrónico a su ordenador. En Outlook Express haga clic en Herramientas, Reglas de mensaje, Correo y use la Regla de Correo Nueva para especificar qué tipo de correo no quiere (véase figura 3.1). Por ejemplo, rechace todos los correos que contengan la palabra *sexo* o *Dinamarca* en el Asunto, o las palabras *amigo* o *tú* en el campo Para. Eche un vistazo a su CORREO BASURA y construya sus propias reglas para eliminarlos sin perder los mensajes que realmente le interesan.

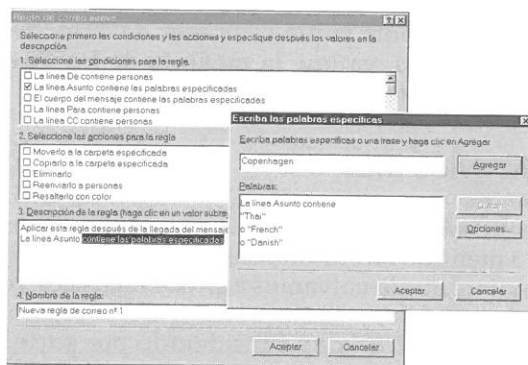


Figura 3.1. Aquí es donde tiene que filtrar el correo en Outlook Express

3.5.4. Filtros en AOL

Haga clic en el botón derecho superior de su ventana AOL y teclee **Mail Controls**. Haga clic sobre el botón que pone **Junk Mail**. Se desplegarán varias opciones. Puede instalar un filtro que le permita recibir mensajes sólo de las direcciones que especifique. O puede informar a AOL de su correo basura para ayudarles a bloquearlo.

3.5.5. Luchar contra estos programas

Si está hasta las narices de este correo basura, puede probar con algunos programas gratuitos para frenarlo.  **SpamHate** puede descargarse en la dirección www.cix.co.uk/~net-services/spam/spam_hater.htm. Esta inteligente utilidad analiza el correo basura e intenta ubicarlo en la dirección por la que entró y enviarlo al remitente. Además, genera un correo de respuesta que puede enviar con un mensaje personalizado (amenaza de represalias o lo que le apetezca). También tiene la característica de evitar que su dirección de correo electrónico pase a formar parte de las bases de datos de las personas que envían este correo basura.

Puede probar con *FakeAddr*. Funciona de forma similar a cuando los pilotos de guerra lanzan hojas de aluminio para distraer a los misiles que se aproximan: arroje miles de cebos al aire y vea lo que pica y lo que no. Fake Addr construye listas de direcciones de correo electrónico falsas que se lanzan a través de Internet para que las recojan los crawler. La idea es que un número considerable de direcciones falsas bloqueen estos robots y que sus bases de datos sean cada vez menos eficaces, ya que la mayoría de las direcciones contenidas no existen.

3.5.6. Un aviso más

Una cosa más antes de cerrar el tema. Los correos basura insertan a menudo en sus mensajes frases de este tipo:

Si no quiere que le volvamos a enviar mensajes, responda a esta dirección...

Puede creer que es muy considerado por parte de ellos eliminarle de su base de datos. Pero recuerde que, al igual que los vendedores de telemarketing, las personas que envían correo basura son, por definición, desconsiderados.

Créame: **Nunca jamás** conteste a este tipo de correo. Responder implica que está vivo, que tiene una cuenta activa y, lo peor, que es un inocente que confía en ellos. Su dirección de correo electrónico se transfiere a una base de datos de *pardillos reafirmados* que se vende como un premio a otros. Esta lista "de oro" contiene sólo aquellas personas que son lo suficientemente inocentes como para responder a un correo basura.

Si contesta, le puedo asegurar que aumentará considerablemente el envío de estos mensajes. Llegado a ese punto, la única solución es cambiar la dirección de correo en su proveedor de Internet y notificar el cambio a quién considere necesario.

Otra forma que tienen de conseguir su dirección de correo electrónico es a través de listas legales como las bases de datos de miembros que tienen los servidores. Estas listas suelen ser de acceso público porque le permite buscar, por ejemplo, la dirección de un viejo amigo con el que se quiera poner en contacto. También verá cómo se envía correo basura a direcciones generadas por el ordenador, creando prácticamente cada combinación posible de caracteres y dígitos y remitiéndolos a esta enorme lista.

La mayoría de los servidores bloquean el correo basura todo lo que pueden. Cuando reciben correo entrante dirigido a miles de direcciones falsas (las listas generadas por el ordenador causan este efecto) el servidor chequea este correo y lo bloquea. Otra táctica es crear cuentas falsas, dejar que el crawler las recoja y bloquear el correo de aquél que intente enviar mensajes a estas cuentas. Es una continua batalla entre gente egoísta y sin ética y aquellos que intentan proporcionar un servicio efectivo, y en el medio de todo inocentes como nosotros: como usted y como yo.

Los grandes motores de búsqueda: localice cualquier cosa en la red

Seguimos hablando de correo electrónico. ¿Cuál es la mejor forma de saber la dirección de correo electrónico de alguien del que conoce su nombre (como un antiguo compañero)? ¿Qué motor de búsqueda proporciona los resultados más rápidos y exhaustivos cuando quiere encontrar páginas Web?

Puede probar con los metamotores Ferret, que le buscan rápidamente los mejores buscadores como Yahoo,

AltaVista, Google, etc. Si Ferret no lo encuentra, es posible que no exista. Y estos programas son gratuitos.

Vaya al sitio www.ferretsoft.com Allí encontrará diferentes programas Ferret: WebFerret, EmailFerret, IRCFerret, PhoneFerret, FileFerret, NewsFerret, InfoFerret y AuctionFerret.

WebFerret es excelente para encontrar páginas Web pero para buscar direcciones de correo debe usar EmailFerret. Todas las utilidades de Ferret están disponibles en el sitio Web de FerretSoft, que es propiedad de Ziff-Davis, quien compró el software y lo ofrece gratis.

EmailFerret busca a fondo la dirección de correo requerida utilizando simultáneamente otros motores de búsqueda, entre los que figuran Bigfoot, Internet@ddress.finder, Switchboard y WhoWhere?. Teclea nombre y apellidos y, de forma opcional, la dirección en la que cree que puede vivir.

WebFerret es una forma estupenda de localizar información en Internet. Tiene otras características ventajosas como listar las páginas según el criterio de relevancia que establezca, permitiéndole las concordancias que quiera, eliminar sitios duplicados y, opcionalmente, filtrar pornografía y lenguaje malsonante.

Conseguir contraseñas y hacer el *Rat Dance*

¿Cómo consigue un hacker entrar en un sistema informático? Existen diversas formas pero, como puede imaginar, el objetivo principal son las contraseñas. La mayoría de los sistemas en red requieren nombre de usuario y contraseña antes de permitir el acceso al sistema. Es más, muchas redes tienen niveles jerárquicos de poder, así que conseguir la contraseña de un nivel alto es el máximo objetivo para la mayoría de los hackers. Algunos niveles de acceso bajos sólo permiten al usuario, por ejemplo, crear nuevos archivos o leer los existentes, pero no borrar archivos o cosas por el estilo.

A otras personas, denominadas superusuarios, se les puede permitir abrir o leer cualquier archivo, pero no borrar cierto tipo de archivos. Sin embargo, en los niveles más elevados, pueden hacer lo que quieran, donde quieran y como quieran. Este nivel está generalmente reservado a gestores del departamento de informática. A los hackers, por supuesto, les encanta conseguir las contraseñas que les permitan este nivel de libertad. Y, por extraño que pueda parecer, estos gestores se dan contraseñas del estilo *accesototal* o *nivelsuperior*. Deberían ser más precavidos.

La verdad es que es más fácil averiguar la contraseña que, por ejemplo, la combinación de una caja fuerte. ¿Por qué? Pues porque la gente parece tener flaca memoria para recordar buenas contraseñas (las mejores son aquellas formadas por dígitos y no caracteres alfabéticos, y lo que es aún peor, palabras).

Generalmente, es el propio ser humano el punto débil dentro del sistema de seguridad. Por ejemplo, hay quien escribe la contraseña en una nota autoadhesiva y la pega en el monitor porque no es capaz de memorizarla. Es facilísimo

colarse en una empresa, dar una vuelta por las oficinas y echar un vistazo a esas notas amarillas donde pone *mi contraseña: pumpkin*.

También en Internet hay muchas utilidades disponibles que craquean las contraseñas. No se lo ponga fácil escogiendo una palabra que se pueda encontrar en el diccionario o evitando usar dígitos.

4.1. Cómo entran los hacker

Los hacker utilizan diversas técnicas para entrar en una red o en un ordenador. Una forma de entrar en un sistema seguro es dejar un *mockingbird*, que es un pequeño programa que intercepta las combinaciones de identificador de usuario y contraseña cuando se teclean y se las envían al hacker.

Fíjese que el trabajo de consulta de un programador es otra forma de entrar en el sistema. El programador-hacker crea una puerta trasera, que es una entrada al sistema que burla la seguridad. Los programadores pueden crear intencionadamente estas entradas por razones tan legítimas como proporcionar a los servicios técnicos una forma analizar el sistema o alguna aplicación. Una puerta trasera es como una entrada secreta o escondida. El programador conoce la clave de esta entrada pero nadie más conoce su existencia.

4.1.1. Vamos a contar mentiras

Utilizar las debilidades humanas para burlar la seguridad es un arte antiguo y aún sigue vivo hoy en día en el ciberespacio. La *ingeniería social* es el término empleado por los hacker para confundir o engañar a la gente y, probablemente, la forma más eficaz de entrar en sistemas protegidos. No consiste en alta tecnología, a no ser que se considere técnico el arte de interpretar, y casi siempre funciona de maravilla.

Una forma de ingeniería social que utilizan los hacker se denomina *spoofing* (engaño). El hacker envía un correo a alguien fingiendo ser un jefe o algún empleado del departamento de servicios informáticos. Le pide una verificación de contraseña pidiéndole que la teclee y la envíe, de forma que "pueda comprobar" su autenticidad. Se maravillaría si supiera con qué frecuencia funciona este truco. La mayoría de

nosotros estamos programados para ser amables y hacer lo que se nos pide, especialmente cuando la petición la realiza un superior.

Otro engaño típico requiere dos elementos de información. Imaginemos que usted es un hacker para ver cómo funciona la cosa. Tiene que conseguir las direcciones de correo electrónico de una persona importante de una organización (fácil) y además el nombre de alguien del departamento de informática de la propia organización. Este segundo elemento no siempre se necesita. Si la organización es lo suficientemente grande, es casi imposible que alguien conozca los nombres de todos los administradores de sistema, así que invente un nombre.

Por ejemplo, supongamos que descubre la dirección de uno de los jefes superiores de ventas, *JakeSims@powertree.net* quien, dado el puesto que desempeña, tiene un alto nivel de acceso a la red, es decir, puede borrar archivos, cambiarles el nombre y manipular el sistema de forma potente.

Le envía este mensaje:

"Hola, Sr. Sims:

El departamento de servicios informáticos ha decidido hoy que hay que asignar nuevas contraseñas. Por favor, espere un momento para introducir la siguiente orden:

Establecer contraseña: U22L5.

A continuación, acepte. Le rogamos que no anote la contraseña. Por razones de seguridad, tenemos que pedirle que la memorice. Gracias por su colaboración.

Randy Thompson, administrador de sistemas".

Más tarde, utiliza esta contraseña y ya tiene acceso al sistema desde un poderoso nivel privilegiado. Si es un hacker bueno y responsable, sólo echará un vistazo para satisfacer su curiosidad (leer memorias de lectura, observar los sueldos de la gente, mirar los resúmenes secretos de los empleados). Pero, si es un cracker malo, comenzará a borrar archivos, copiarlos, moverlos e irá arrasando todo lo que encuentre a su camino. Y si es un whacker, a saber lo que hará. Quizás acabe con toda la empresa.

Si trabaja ya en una empresa, puede averiguar muchas contraseñas leyendo las notas autoadhesivas. Recuerde que mucha gente tiene que escribir su contraseña para no olvidarla. Se lo adviertan o no, estas personas tienen la virtud de dejar sus contraseñas a la vista de cualquiera que casualmente pase por allí.

4.1.2. Hola, soy nuevo en la oficina

Otra forma muy habitual de ingeniería social que rompe el sistema de seguridad es dirigirse al administrador de sistemas y fingir que se es un nuevo empleado.

"Hola, me llamo John Jette y he empezado a trabajar hoy en el departamento de contabilidad. Siento molestarte, pero llevo 15 minutos intentando engancharme a la red (risilla tonta, ji, ji, ji) y tengo que pedirte ayuda. Me gustaría que me enseñaras todo esto. ¿Puedes darme una lista en la que se me indique paso a paso lo que tengo que hacer para entrar?"

Este truco puede fallar por diversos motivos, siendo el más lógico que dijese *"¿Por qué no le pide ayuda a su administrador?"* Pero, la mayoría de los problemas tienen su solución y un buen ingeniero social lo haría una media hora antes de que cerrase la empresa y diría. *"No hay nadie en mi departamento ahora. Por favor. Tiene que ser fácil"*.

4.1.3. El timo de la equivocación del experto

Este engaño se basa en la sorpresa. Llama por teléfono y pican como nada.

"Hola, soy Jenny Tandem del departamento de informática y estamos chequeando de nuevo todos nuestros sistemas. ¿Podrías dedicarme cinco minutos para analizar el throughput del sistema? Muchas gracias. Revisaré tus entradas paso a paso. Acabo en un par de minutos. Sí, gracias, vale. Ahora entra. ¿Qué has tecleado? Sí, lo he visto perfectamente. Bien. Sigue entrando. ¿Qué has tecleado? Exacto, eso es. Ahora escribe tu contraseña. ¿Qué has puesto? Vale, correcto. ¡Genial! Es todo lo que necesitábamos. El sistema funciona perfectamente en tu estación de trabajo. Gracias de nuevo por tu colaboración".

4.2. El problema de las contraseñas

Aunque son necesarias, las contraseñas tienen grandes carencias. Se supone que la gente debería recordarlas así que el empleo de una secuencia de números, la más difícil de craquear, es infrecuente. ¿Por qué? La mayoría de nosotros sólo podemos recordar siete dígitos en un número de teléfono y aún así, tenemos problemas con eso.

Una palabra, como *Félix*, el nombre de su gato, es más fácil de recordar que un número, pero también es más fácil que descubrirla usando las técnicas por la fuerza bruta de los hacker. Por ejemplo, ciertas combinaciones de letras son más comunes que otras. Este tipo de técnicas (por ejemplo, intentar todas las combinaciones posibles hasta que salte una) evita combinaciones raras, como, en el caso del inglés dos *u* seguidas, ya que sólo existe una palabra en este idioma que las contenga (*vacuum*), por lo que no se intentará esta combinación.

Quizás no se sorprendería al saber que, sin haber reglas ni normas, la mayoría de las personas eligen su propio nombre, el nombre de su pueblo, palabras como amor, papá, tierra, mundo. Ya se hace una idea. Si tiene paciencia y desea pasar algún tiempo intentando conseguir contraseñas, es probable que consiga entrar en varias redes sin un gran esfuerzo. Fíjese que la mayoría de las actividades de los hacker no requieren una inteligencia especial ni gran cualificación técnica. La cualidad esencial es tener mucha paciencia.

Compárelo con la pesca. Se sienta y lo intenta una y otra vez, arroja, recoge, arroja, recoge, y así sucesivamente. Simplemente el ritmo y la repetición resultan relajantes. Y entonces ¡zas! Un tirón. Lo ha pescado. De repente, brota de la tranquilidad un chorro de adrenalina que premia la fantástica sensación de la paciencia. No es ninguna sorpresa que el hacking, al igual que la pesca, llegue a causar adicción.

Hay muchas utilidades para craquear contraseñas que se pueden bajar de Internet. Algunas de ellas son muy efectivas. Por lo tanto, si elige una contraseña que se pueda encontrar en un diccionario o un nombre común o cualquier otra palabra habitual, casi ni le merece la pena tenerla. No tiene ninguna protección usando contraseñas como "árbol" u "obligación". Utilice números y la combinación más larga posible e intente evitar el uso de palabras comunes.

4.3. Abrir el maletín misterioso

Hace unos años me encontré un maletín de piel muy caro, marca Hartmann. Lo sacudí pero no logré saber si estaba vacío.

Tenía un cierre de seguridad con tres contadores. Me convencí de que podía intentar abrirlo y me dije a mí mismo que tenía un 50 por ciento de posibilidades de abrirlo si probaba la mitad de las posibles combinaciones.

Primero pensé que abrirlo sería muy problemático, me llevaría un montón de tiempo y sería muy aburrido.

Empecé a girar los contadores, y después de un par de minutos, empecé a sentir la fascinación de la pesca, la calce y pasatiempos similares.

Tras un suspiro, empecé mi trabajo. Puse los tres contadores en 0 0 0. Intenté abrir. Nada. Probé con 0 0 1. Nada. Enseguida cogí un ritmo y me sobrevino una especie de calma. Cambio de contador, abrir, cambio de contador, abrir... Finalmente, allá por el 700 cedió el botón de apertura y dos barritas se abrieron haciendo un hermoso clic, clic.

Pero los fajos de billetes de 10.000 que esperaba encontrarme dentro no eran tales, y en su lugar había una calculadora barata, un par de bolígrafos y un bloc de papeles. Pero, por primera vez en mi vida, entendí el placer y la excitación que puede sentir un hacker.

En este capítulo, hemos jugado a ser hacker y hemos intentado entender sus métodos y cómo piensan. Más adelante, nos quitaremos el "sombrero negro" y nos cambiaremos de bando, viendo cómo se puede uno proteger de los hacker de contraseñas. Sin embargo, si quiere conocer un poquito más el lado oscuro, puede navegar por grupos de noticias donde participan hackers y ver sus formas de pensar. Vea la sección al final de este capítulo titulada "Dónde recolectan los hacker".

4.4. El *Rat Dance*

Aquí tiene otro elemento más acerca de los hackers que le puede interesar. El *Rat Dance* (baile norteamericano)¹ es otro de esos fantásticos términos que me parecen absolutamente irresistibles. Prometo que es la última vez que utilizo un título equivocado en este libro. No pude evitar usarlo, aunque no está directamente relacionado con el tema de las contraseñas.

Hacer el *Rat Dance* se refiere a que el hacker obtiene resultados diferentes de los que buscaba, pero no por ello dejan de ser valiosos o interesantes. Por ejemplo, supongamos que un hacker intenta localizar el directorio que contiene las nóminas de los empleados. Podría intentar conectar su ordenador a ese directorio, escribiendo repetidas veces las distintas posibilidades:

NÓMINAS

Directorio no válido

¹Nota del Traductor.

FINANZAS

Directorio no válido

PERSONAL

Directorio no válido

CONTABILIDAD

Directorio no válido

CONTABLE

Directorio no válido

CONT

¡Bingo! El hacker consigue el nombre de directorio **CONT**. Sin embargo, estuvo haciendo el *rat dance* porque este es un directorio que contiene todas las contraseñas del sistema pero no los datos de las nóminas. El resultado es válido, muy interesante, pero no el buscado inicialmente.

Dónde recolectan los hackers

Hay varios grupos de noticias activos en los que puede encontrar:

1. *Hackers debatiendo acerca de las últimas técnicas.*
2. *FBI y otros organismos leyendo los debates.*
3. *Whacker y cracker registrándose.*
4. *Futuros hackers ocultándose.*
5. *Autores enfadados de libros de informática echando un vistazo.*
6. *Otras personas interesadas en tácticas para entrar en sistemas, realizar phreaking a empresas, conseguir software comercial gratuito (warez) y otros temas relacionados con este mundo de los hacker.*

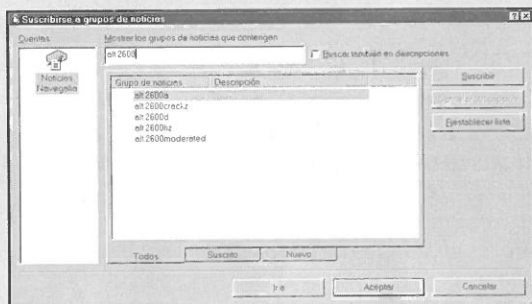
¿Qué necesita craquear de un archivo de Word cifrado? Puede encontrar programas de craqueo de cifrado y los sitios Web donde se venden. Esta información, y mucha más, se encuentra disponible en Hacker Central, diversos grupos de noticias agrupados bajo un grupo general denominado alt.2600.

Yo no utilizo warez, ni oculto mis ID, no uso cheques arrancados de un talonario ajeno ni como bistecs del super que me he llevado metidos en los pantalones. Pago lo que necesito, y lo que no me puedo permitir, intento no desearlo. Pero le echo un vistazo de vez en cuando a este mundillo y

a los cracker y whacker atraídos por él. Para mí, seguir alguno de los debates es como leer las últimas noticias de la gente rica y famosa; todos parecen haber perdido el rumbo ético, pero hay que reconocer que son interesantes.

La mayoría de los whacker y cracker son más perseverantes que inteligentes, pero sus debates suelen ser interesantes. La fascinación con estilo por encima de la sustancia. Esa arrogancia apenas reprimida. Los interminables trucos. Prácticamente no se distinguen de Madonna desde un punto de vista psicológico.

Para darnos una vuelta por el lado salvaje, escoja en el menú de Outlook Express Herramientas, Grupos de noticias (o el equivalente en Netscape). A continuación, escriba alt.2600 en el cuadro Mostrar los grupos de noticias que contengan. Verá el siguiente gráfico:



En los grupos de noticias que aparecen en los gráficos, estará codo con codo con los hacker, whacker y cracker en todo su esplendor; en estos grupos, las travesuras son el orden del día. Esta gente tiene lo que el orientador de mi colegio llamaba tendencias.

Verá un montón de z o zs en la versión inglesa: alt.2600.cardz, alt.2600.codez, alt.2600.crackz, alt.2600.hackerz, alt.2600.phreakz, alt.2600.programz y, por supuesto, alt.2600.warez.

Eche un vistazo a los diversos grupos y verá un montón de cosas. Estará bien acompañado. La CIA y otros merodean estos grupos. Y, en su lugar, evitaría mandar mensajes por razones obvias.

La "Venus atrapamoscas" y otros anti-hacker

Por cada hacker que intenta entrar, hay un administrador de sistemas al otro lado intentando construir sólidas paredes para salvaguardar la información. Pero ya ve el problema: para que la información estuviera completamente segura, habría que sellarla tan herméticamente que nadie podría acceder a ella. Para ser práctica, una base de datos (agrupamiento de información en un sistema) debe ser accesible prácticamente a diario y a muchas personas.

Por todo el país, los ancianos se enfrentan a la "paradoja de las paredes seguras". Han terminado de pagar su hipoteca, pero ahora sus ingresos mensuales son infinitamente menores. Tienen que quedarse en esa casa porque no se pueden permitir una mejor y esta ya no vale nada. ¿Por qué? Porque durante los treinta años que han estado viviendo allí, el vecindario se ha colapsado. Lo que antes era un enclave de clase media con una bonita alameda se ha convertido en una maldita casa podrida de los suburbios, con coches abandonados ardiendo y disparos durante la noche.

¿Qué pueden hacer? Tras haber sufrido robos, deciden poner rejas en las ventanas. Estas tienen el inconveniente de que dificultan la salida de la casa en caso de incendio. Pero tienen más miedo de que los ataquen los descerebrados zombis que del fuego. Al fin y al cabo, los incendios son poco frecuentes, pero casi cada noche pueden escuchar como los muertos vivientes golpean en su ventana, intentando encontrar la manera de entrar. Así que el anciano matrimonio se enfrenta como puede con la paradoja inherente a las rejas de seguridad: las hacen tan recias que aprisionan todo lo que se queda dentro.

5.1. La defensa de las empresas

Trataremos la defensa de los ordenadores personales más adelante. Este capítulo continúa con la línea de los anteriores: los peligros que los hacker causan a los sistemas de las empresas y las soluciones que las empresas norteamericanas han usado para frenar la oleada de delincuencia cibernética. Aquí tiene algunas de las técnicas más exitosas.

5.1.1. Cebo y captura

Denominaciones como *caja de hierro*, *cortafuegos* o *Venus atrapamoscas*, son las utilizadas para referirse a utilidades especiales para atrapar a un cracker que está entrando desde una conexión remota. La idea es proporcionar al hacker un acceso limitado al sistema, sin que este límite sea demasiado obvio, ofreciendo información interesante (conocida como archivos cebo) para mantener al intruso el tiempo suficiente dentro del sistema como para después poder dar con él.

A veces, una caja de hierro está compuesta de un sistema operativo ficticio, una concha falsa que parece algo real, completo, pero que restringe al intruso de formas que no son fácilmente perceptibles. Se pueden incluir archivos falsos pero interesantes para atraer al hacker. Otras veces, una caja de hierro está configurada para detectar movimientos específicos o determinados identificadores y contraseñas. La idea general es detectar y luego retener al intruso, mantenerlo un rato dentro del sistema. Funciona de un modo similar al intento de localización de llamada telefónica, en el que hay que intentar mantener la conversación mientras la policía averigua desde dónde se realiza.

Un cortafuegos (Venus atrapamoscas) es un ordenador configurado especialmente para controlar las llamadas entrantes en la red desde el exterior. Esta máquina, que recibe una atención especial por parte de los administradores de sistemas, mantiene un alto nivel de seguridad en comparación con otras del sistema. Es el guardián de la puerta y no contiene archivos importantes, si bien puede almacenar archivos cebo. Generalmente, el cortafuegos tiene múltiples líneas de entrada, pero sólo una principal supervisada que va a la red de la empresa. Por supuesto, el concepto de cortafuegos ha ido en aumento a lo largo del tiempo para ser

algo más que un dispositivo de vigilancia. Incluso podrá encontrar software descrito como "programas para corta-fuegos personales".

5.1.2. Vigilancia constante

Algunos expertos en seguridad aseguran que es un tema demasiado complejo para entendidos en informática que trabajan desde sus casas. El departamento de informática (IS) puede instalar un cortafuegos, requerir frecuentes cambios en contraseñas sofisticadas (incluidas combinaciones de dígitos y caracteres alfabéticos) y utilizar un complejo cifrado para proteger la información de un ataque externo. Pero estos enfoques clásicos no son suficientes. Según autoridades profesionales de la informática lo que se necesita es ayuda externa.

No puede evitar que la gente intente romper sus defensas del mismo modo que no se puede evitar que los mosquitos choquen contra las ventanas y lleguen a entrar en casa. No importa lo segura que parezca su protección: es imposible cerrar todos los agujeros de su pantalla. Los hackers disfrutan con el reto de entrar en sistemas informáticos y, recuerde, que suelen ser gente muy paciente que prueba una y otra vez hasta que algo funciona.

Los expertos dicen ahora que lo que se debe hacer es identificar las irrupciones en cuanto sucedan y tomar medidas inmediatamente para enfrentarse a esas entradas. No es demasiado práctico contratar expertos durante un par de días para realizar trabajos internos. En vez de eso, se necesita el equivalente a una alarma casera y una empresa externa de control conectada al sistema, preparadas para responder con rapidez (a las tres de la mañana, si a esa hora salta la alarma).

El punto más débil del típico sistema de defensa contra hacker en una empresa es que no están diseñados para estar siempre alerta y preparados para contestar. Por ejemplo, si la incursión empieza después de que la gente del departamento de informática se haya ido a casa ¿quién se queda allí para solucionar el problema?

5.1.3. La defensa "Interfaz de 10 dedos"

Como se puede imaginar, ha habido reacciones extremas a los ataques de los hacker. El tipo más radical de seguridad de red, cerca de coger el hacha para cortar las líneas de telé-

fono, es la denominada Defensa de Interfaz de 10 dedos. ¡La red se desconecta físicamente del servidor! Una terminal contiene todos los módems o puertos públicos que aceptan toda la información entrante (en el medio, con intentos de uno o dos hacker pesados). Pero la red de trabajo, u otro sistema, no está físicamente conectada a este ordenador. Un trabajador humano observa la información entrante (Output) en el ordenador y, a continuación, teclea el resultado (Input) en otro ordenador que sí está conectado a su sistema. Este método tiene muchas desventajas como método eficiente de transferencia de datos. Hablando en plata, es lentísimo.

Otra respuesta todavía más inefectiva por parte de los proveedores de sistemas operativos y de software puede describirse como *seguridad mediante oscuridad*. Cuando los distribuidores de sistemas operativos descubren un agujero en sus defensas, o bien lo ignoran, esperando que nadie se dé cuenta o esperan que los hacker no se aprovechen de él. Sin embargo, esta táctica de agachar la cabeza bajo el ala es cada vez menos frecuente y cuelgan en Internet soluciones que se pueden descargar rápidamente. Poco a poco, la gente va aprendiendo la vieja lección: la sinceridad es la mejor política. Se sufren menos consecuencias a largo plazo si se admiten los errores y se ofrece ayuda para solucionarlos que si se intenta evitar enfrentarse a ellos.

5.2. Soluciones prácticas para empresas

A menos que sea una estrella de cine, no creo que contrate sus propias fuerzas de seguridad las 24 horas del día. Los guardaespaldas y seguritas sólo se los pueden permitir los ricos. Sin embargo, mucha gente protege su hogar con un sistema de alarma antirrobo y lo conecta a la comisaría de policía local o a un servicio de vigilancia continuada.

Del mismo modo, pocas empresas instalan su propio pequeño ejército para proteger sus almacenes y oficinas, sino que subcontratan este servicio a una empresa de seguridad. Es mucho más barato y sencillo subcontratar este tipo de servicios que intentar crearlos dentro de la propia empresa. Considere cuántos servicios se subcontratan: desde servicios de conserjería hasta el traslado del dinero desde la tienda al banco. Del mismo modo, subcontratar la seguridad informática de la empresa es inteligente y rentable.

Recuerde que el hacking es un entretenimiento que se lleva a cabo a altas horas de la madrugada o durante el fin de semana. ¿Se le va a pedir al equipo de informáticos, ya bastante ocupado, que dejen las lucecitas encendidas en el departamento 24 horas al día, todos los días del año? ¿Se supone que tienen que estar allí alerta todo el tiempo? Y más allá de esa vigilancia, ¿puede esperar que sean expertos en todos los aspectos para dar una respuesta de defensa inmediata a un hacker y reparar los daños? Su bandeja ya está bastante llena. Son expertos en el funcionamiento de las aplicaciones clave de la empresa, en los sistemas de facturación conectados a la base de datos de los clientes y otras muchas funciones críticas de la empresa. No puede esperar que comiencen a trabajar por turnos y añadir nuevas categorías de especialización a sus ya atareadas vidas.

Como los ataques de los hacker a una empresa determinada son esporádicos, es más práctico subcontratar una empresa de seguridad que está atenta las 24 horas del día, está a la última en temas de virus y técnicas de entrada de los hacker y puede reaccionar de forma efectiva, reducir el daño y llevar a cabo reparaciones. Esta gente son profesionales. Sobre todo, es altamente efectivo contratar a una empresa que se encargue de controlar varias empresas a la vez, dada la infrecuencia de ataque de un hacker. Es casi imposible que ataquen simultáneamente a sistemas de diferentes compañías.

Los expertos afirman que se necesitan de cuatro a cinco personas para ocuparse de la seguridad 24 horas al día, siete días a la semana. Y, si empieza el ataque de un hacker, deberían incorporarse al equipo otras cinco personas más. Este es el mínimo de personal que se necesita para llevar a cabo una reacción inmediata y efectiva contra el hacker. Recuerde que cada una de estas personas es un experto cualificado. ¿Está una empresa dispuesta a contratar personal a este nivel para enfrentarse a un problema que probablemente surja, como mucho, una vez al año?

Por supuesto que depende del tamaño de la empresa el que sufra más o menos ataques. Pero lo más importante es el trabajo que se desempeñe. Es más probable que interesen más a los hackers empresas que desempeñen actividades del ámbito político y cultural por la información que puedan obtener (como contratos de defensa, transacciones económicas, información financiera) que, por ejemplo, una empresa de comida para perros.

5.2.1. Envíe a los Marines

Algunas compañías han adoptado una postura diferente que consiste en contratar a personas ajenas a la empresa para simular el ataque de un hacker. La idea es ver la extensión que este puede tener, sin llegar realmente a robar información o destrozar el disco duro. ¿Consiguen estos impostores entrar? ¿Pudieron haber ejecutado órdenes de eliminación de archivos?

A este tipo de hacker de alquiler se le llama fisgón. Un fisgón es alguien al que se le paga para que intente entrar en un sistema, probar sus defensas. A un grupo de fisgones se le denomina equipo de simulación de penetración en el sistema (que proviene de un término militar utilizado para describir a un grupo de soldados que intentan físicamente burlar los sistemas de seguridad).

Esta técnica puede ser útil a la hora de identificar puntos débiles pero no es sustituto de la vigilancia que pueden proporcionar los servicios de vigilancia las 24 horas del día.

5.2.2. Consideremos la posibilidad de un seguro

Se puede considerar una alternativa a intentar tener todo tipo de defensas contra hacker. En vez de preocuparse de instalar software y hardware especial, de contratar a otros y otras medidas de protección, ¿qué tal asegurarse contra los perjuicios causados por un hacker? Empresas como Counterpane Internet Security pueden llegar a darle hasta 20.000.000 millones de pesetas, empresas que cubren, al menos económicamente, la pérdida o sustracción resultante de una irrupción en su sistema informático. Échele un vistazo en www.counterpane.com.

5.2.3. Otra vez la paradoja de las rejas en las ventanas

Demasiada protección causa problemas, como bien le podrá decir alguien al que, queriendo cruzar el Canal inglés a nado, le hayan obligado a ponerse grasa de ganso para mantener el calor o los pesados chalecos antibalas de acero que llevan a veces los policías. Algunos peces con pequeñas púas se sienten atraídos por la grasa de ganso y pican en las

piernas. Es difícil perseguir a delincuentes vistiendo ropas de acero. De nuevo la vieja paradoja: la protección parece limitar la libertad.

Los cortafuegos están diseñados para permanecer entre el hacker y el sistema de la empresa. Pero, ¿qué pasa con las necesidades legítimas de los empleados que necesitan hacer el camino inverso, es decir, atravesar el cortafuegos para entrar en Internet? ¿Pueden hacerlo? Apueste.

Supongamos que es un empleado normal que está usando el Microsoft Internet Explorer y quisiera especificar un HTTP (Hyper Text Transfer Protocol Proxy) antes de entrar en Internet. Este proceso requiere conseguir primero una dirección IP (Internet Protocol) para un proxy. Le pregunta a alguien del departamento de informática cómo conseguirla y cómo instalarla usando el **Panel de Control** de Windows.

Si esto no funciona, pregunte o ruegue si es necesario a la persona encargada en el departamento de informática cómo puede acceder a través del cortafuegos. Se supone que es para evitar que entre la gente mala en el sistema, no para evitar que buenas personas no puedan acceder a la cantidad de información útil disponible en la World Wide Web.

Este es el principal problema: las redes están diseñadas para facilitar el compartimento de los datos. Por lo tanto, la idea de la privacidad de datos es totalmente opuesta al propósito de una red de trabajo. Esto no quiere decir que no pueda alcanzar un punto medio entre ambos aspectos necesarios y conflictivos entre sí. Cada empresa debe calcular el intercambio y decidir cuántas ventanas cubrir con rejas.

Una de las soluciones a este conflicto son los distintos tipos de cifrado. La idea aquí es que, si no puede mantenerlos alejados del sistema, puede dejar al menos datos importantes tan bien cifrados que sea imposible para ellos entender qué es lo que han robado. Buenos candidatos para el cifrado son listas de contraseñas, datos de los empleados, proyectos y cualquier cosa que interesaría a intrusos.

Nos introduciremos en el fantástico mundo de la criptografía en la segunda parte. Es un arte antiguo que comenzó ya antes de que los primeros faraones ordenaran cortar la lengua de los esclavos que trabajaban en sus tumbas secretas. Pero la criptografía ha tomado una nueva dirección fascinante ahora que los ordenadores posibilitan la codificación de la información de formas muy complejas, y, en el otro lado, muchas soluciones complejas para intentar craquear la información codificada y devolverla a su estado original.

5.3. Pensar en todas las posibilidades

Los trabajadores de la era de la información deben reconocer que el daño puede venir de dentro y fuera de la empresa. Empleados enfadados o tarados pueden hacer el mismo daño que un hacker y tienen la ventaja de estar dentro de los límites del cortafuegos.

Esta amenaza interna, junto con los trabajadores que puedan causar problemas simplemente por torpeza, es la razón por la que se establecen distintos niveles de seguridad dentro de una red de trabajo. Recuerde que algunos empleados del departamento de informática conocen todas las claves, mientras que al final de la cadena, otros trabajadores tienen un acceso muy restringido a ciertos directorios y, muchas veces, de sólo lectura.

En el siguiente capítulo, finalizaremos este viaje a través de las opciones disponibles para los gestores actuales, que tienen que encontrar el punto intermedio entre guardar los archivos en un búnker y dejar la red completamente abierta para que, en cualquier momento, entre cualquiera que quiera ver algo.

Buscar el punto intermedio

Si se toman muchas medidas de seguridad, tendrá problemas para acceder a la información que pretende proteger. Cuanto más endurezca las medidas, reducirá al mismo tiempo la velocidad de transmisión de la información y con ello disminuye la productividad. Si se excede, las personas que deberían tener acceso a la información no pueden obtenerla en un tiempo razonable, si es que consiguen acceder a ella.

Sin embargo, si es demasiado liberal y no toma medida alguna, entonces sus rivales verán sus planes para el futuro, sus virtudes operacionales, los puntos débiles que pueden explotar y leer sus tácticas empresariales. Hackers desequilibrados o ex-empleados enfadados pueden darse un paseo por su red, frenarla o, incluso, destruir datos. Y no olvide nunca la amenaza adicional de empleados desquiciados o enfadados. Ellos sí que están dentro, y, por lo tanto, la mayoría de los sistemas de seguridad, como los cortafuegos, no ofrecen protección.

6.1. Primeros pasos para crear un lugar de trabajo seguro

Es el momento de resumir los pasos que una empresa puede dar para proteger de amenazas internas y externas la integridad de sus sistemas de información.

6.1.1. Invertir la ingeniería social

Si los empleados lo toleran, es una buena idea comenzar a tomar sus medidas de seguridad a partir del vínculo más

vulnerable: los trabajadores. Identifique dónde se almacena la información más importante y quién tiene acceso a ella. A continuación, reúnalos para hablar de las distintas formas de ingeniería social que utilizan los hacker para conseguir las contraseñas.

Cuénteles la historia de llamadas urgentes del FBI o personal de servicios, o fulanito de tal del departamento de informática para pedir las contraseñas. *¡No proporcione nunca una contraseña u otra información de seguridad por teléfono a alguien que no conozca personalmente! Y explíqueles los trucos de ingeniería social que hemos descrito.*

Puede dar también los siguientes pasos:

1. Cambie las contraseñas de forma regular, con la frecuencia suficiente para establecer diferencias pero sin molestar a los trabajadores o peor, causar que las apunten en cualquier parte.
2. Explíqueles por qué es importante la seguridad y pídale que colaboren con ello. Explíqueles los diversos niveles de acceso al sistema, la explicación de esta división y los pasos adecuados para asegurarse de que la información importante se guarde en los archivos adecuados.
3. Pídale a la gente que escriba sus contraseñas en un papel y que lo escondan en un libro o detrás de un jarrón.
4. Revise la lista de contraseñas y rechace las opciones fáciles de averiguar, por ejemplo, *pizza*, *cuenta*, *máquina*, *María* (o cualquier otro nombre o iniciales), *basededatos*, etc.
5. Insista en que las contraseñas deben exceder los seis caracteres.
6. Exija que cada contraseña contenga al menos un dígito.

6.1.2. Desarrollar y mantener una política de seguridad

Es importante percatarse de que se debe realizar un análisis de seguridad si se trabaja con información relevante en la empresa. Y casi siempre la hay. Por lo menos, no querrá que los datos de evaluaciones y sueldos de sus empleados anden de boca en boca. Tampoco querrá invitar a los gusanos a su sistema provocando que cada vez vaya más y más lento hasta que se queda paralizado.

Por lo tanto, dependiendo del tamaño de su empresa, de sus recursos financieros y del grado de amenaza, establezca una política de seguridad constante que mantenga el equilibrio perfecto entre el escollo que supone una seguridad excesiva y la difícil situación de exponer el sistema a los hacker.

Existe la posibilidad de contratar especialistas, como Fujitsu (www.fsba.com/services/monitoring.html) o un servicio similar, para chequear el sistema, localizar los puntos vulnerables y sugerir soluciones. Verá que los servicios de seguridad pueden proporcionar un primer análisis de sus necesidades, pero también ofrecen chequeos adicionales posteriores, supervisión y servicios de reparación. Por ejemplo, entre los servicios de Fujitsu figuran:

1. Una supervisión continua sobre un posible intrusismo que incluye respuestas mediante avisos por correo electrónico o una intervención directa ajustando su cortafuegos.
2. Un chequeo periódico del cortafuegos para ver si hay algún punto vulnerable en sus defensas.
3. Actividades de entrada en el sistema con informes de todos los movimientos en la red (intentos de entrada de hacker, tráfico no autorizado, etc.).
4. Detección continua de virus y limpieza. Se le notifica inmediatamente el problema por correo electrónico y papel.
5. Filtrado del contenido de Internet restringiendo la entrada de los trabajadores a sitios específicos o categorías de sitios.

Una vez que tenga unas líneas establecidas, sígalas. Réviselas para sugerir cambios y asegúrese de que se están implementando. Haga que su política de seguridad sea responsabilidad de una sola persona con el estatus suficiente como para que se lleven a cabo las normas. No atribuya esta función a un miembro inexperto del departamento de informática.

La seguridad puede llegar a ser cara: nuevo hardware, seguramente personal nuevo y nuevo software, o quizás contratar consultores externos o servicios de supervisión para enfrentarse al problema. Puede querer implementar sus planes de seguridad paso a paso, empezando con lo que la mayoría de los profesionales consideran la adquisición básica: un cortafuegos. Se estima que los cortafuegos pueden protegerle de un 10 por ciento de los ataques y muchas empresas creen que lo único que necesitan es un cortafuegos bien hecho. Hablaremos de esta indispensable protección posteriormente.

Después de haber instalado su propio cortafuegos y ejecutarlo, puede adquirir protección adicional como servicio de supervisión, software de cifrado, etc.

6.1.3. Comprobar la identidad

Un paso relativamente barato para asegurar el sistema es usar las contraseñas de forma correcta, como hemos descrito anteriormente. Además de las contraseñas, puede tomar medidas adicionales para asegurarse de que sólo entra en el sistema quien tiene que hacerlo.

Puede darle a cada empleado una contraseña y una tarjeta especial. Esta es una tarjeta de identificación que se puede usar una vez insertada la contraseña correcta. Si usa una tarjeta inteligente, puede reprogramarse cada vez que se utilice tomando la siguiente dentro de una serie de claves cifradas (la clave puede cambiar cada vez que el usuario pasa la tarjeta). Esto aumenta considerablemente la seguridad. Pero incluso si la tarjeta permanece igual y proporciona la misma información cada vez que se use es mejor que sólo una contraseña. La técnica tarjeta-contraseña combina el poco fiable uso de contraseñas (en las que el usuario debe recordar la contraseña y procurar no perderla) con un uso relativamente fiable de la clave física, en la que el usuario ni siquiera sabrá qué clave cifrada hay en la tarjeta. Además, no podrán dársela a aquellos hacker que practiquen la ingeniería social llamando y fingiendo ser la policía.

También puede repartir llaves de metal que deben ser usadas en una terminal de ordenador u otro aparato protegido para conectarlo a la red. Finalmente, la solución más segura y más cara para la identificación del personal es usar máquinas que puedan reconocer retinas, firmas, huellas dactilares, rostros o voces. Ya lo habrá visto en alguna película: un rayo láser que se desliza por el ojo de alguien hasta que las puertas deslizantes de la CIA se abren con un sonido ahuecado.

6.1.4. Túneles, privacidad virtual y otras formas de autenticar las comunicaciones informáticas

Los seres humanos no son el único talón de Aquiles. Tiene que tener en cuenta que los ordenadores hablan unos con otros (¿es muy parlanchín el suyo?).

Cuando abre sus puertas a Internet, puede decirse que las abre en el sentido casi literal de la palabra. Como verá más adelante, las conexiones continuas a Internet son una verdadera preocupación ... ping, ping, ping, los hacker y sus robot llaman continuamente esperando encontrar un hueco por el que colarse. Por supuesto no quiere que su compañía tenga ese hueco.

Se espera que en el futuro esté disponible una solución a los puntos débiles de Internet mediante el establecimiento de un "túnel". Esto significará que su ordenador y otro en cualquier parte del mundo puedan utilizar una técnica de autenticación para identificarse entre ellos y emplear, a continuación, un cifrado de forma que ningún otro ordenador de Internet pueda tener acceso.

En la actualidad, los comités de estandarización debaten acerca del Protocolo Tunneling Nivel 2 (*Level 2 Tunneling Protocol*). Este posibilitaría la creación de canales privados, como si fuesen conductos, para unir dos puntos distantes sin permitir la entrada desde otro punto de Internet (el otro punto sería, en este caso, el rabioso hacker).

Una vez creado el túnel, puede implementar una Red Virtual Privada (VPN, Virtual Private Network). Esto implica que puede usar la tecnología para imitar la seguridad ofrecida actualmente por una Red de Área Privada, la alternativa más cara.

¿Será Internet capaz de funcionar con los túneles? ¿Y tendrán los túneles la misma eficiencia que los mensajes cifrados en la comunicación telefónica entre espías? El tiempo lo dirá.

6.2. Cortafuegos para todas las necesidades

Actualmente, existen varios tipos de cortafuegos al igual que formas de "provocar incendios". La idea en todos es la misma: proteger su información del desastre. Para entender la protección que ofrecen los cortafuegos, examinaremos primero las formas en las que la información se mueve por las redes y por Internet.

6.2.1. Nivel a nivel

Siempre hay organizaciones que establecen, o lo intentan, estándares para que el resto de los humanos los siga-

mos. Existe un comité en Estados Unidos, llamado el World Wide Web Consortium (W3C) que intenta imponer normas a la gente que programa en HTML y a los navegadores que aceptan este lenguaje (este comité ya ha tenido algunos problemas para convencer a los propietarios de los navegadores). El comité americano *Institute of Electrical and Electronic Engineers*, IEEE, es otro ejemplo. Ha contribuido a varios estándares electrónicos además de a temas relacionados con la informática como la criptografía.

Remontándonos a 1983, la Organización Internacional de Estándares (ISO) describió como se deberían estandarizar las telecomunicaciones (ordenadores hablando con ordenadores vía telefónica en aquellos tiempos). Crearon lo que se denominó modelo interconexión de sistemas abiertos para la transmisión de datos. Estaba formado por distintos niveles, tal y como los denominaron. El nivel básico es el físico: módem, cables, chips y cualquier otro aparato electrónico que transmita datos. Por encima de este hay distintos niveles compuestos de varios tipos de software con varias tareas:

- **Nivel de enlace:** Es el software que chequea la integridad de la transmisión de datos. ¿Llegó la información al destino remoto sin errores? Este nivel también es responsable del direccionamiento único de los aparatos en la red.
- **Nivel de red:** Es el software que conecta distintas redes, incluso si no son del mismo tipo (direccionamiento y filtrado).
- **Nivel de transporte:** Es el software que une los paquetes de datos cuando llegan al punto receptor de la transmisión. Este nivel también se encarga de avisar de errores de chequeo para asegurar la integridad de la información.
- **Nivel de sesión:** Es el software que toma algunas decisiones importantes en la transmisión de un archivo: ¿qué código de información se va a usar? (Existen diversos códigos que se usan actualmente, como ASCII, Unicode, etc.) ¿Usará la información enviada métodos de transmisión simplex, duplex u otros? También es el nivel encargado de la conexión y desconexión.
- **Nivel de aplicaciones:** Es el software con el que interactúa cuando está enviando información en Internet: Netscape's Communicator, Internet Explorer, Outlook Express, FTP, Eudora y los demás. En este nivel, pueden protegerse las aplicaciones mediante el

rechazo a una falsa contraseña. Los documentos y bases de datos también se pueden proteger con contraseñas. Nivel a nivel.

Qué son los paquetes

Tiene que entender el concepto de paquetes para entender el concepto de Internet. Un archivo se divide normalmente en pequeñas partes, denominadas paquetes, antes de ser enviado a través de Internet. Estas pequeñas partes se transmiten de forma individual y no necesariamente a través de la misma ruta (quizás un servidor en Italia está menos ocupado que uno de Chicago en el momento en que uno de sus paquetes está siendo enviado a su destino final, Atenas). Los paquetes se usan para equilibrar la velocidad a la que se envía la información, de forma que todo el mundo obtenga resultados similares. Por ejemplo, si un gran banco neoyorquino envía a través de Internet todas las transacciones del día a su central en Charlotte, esta información se divide en muchos pequeños paquetes. Quizás su pequeño correo electrónico consiga viajar con el primer y segundo paquetes. De esta forma, su correo no tiene que esperar a que el gigantesco archivo del banco termine de descargarse lo que realmente podría ralentizar su pequeña transmisión.

Cuando todos los paquetes lleguen a su destino, el nivel de transporte se asegura de que son reconstruidos en el orden adecuado.

Uno de los aspectos más ventajosos de dividir en estos niveles las tareas de enviar y recibir información es que le permite visualizar las tareas y ver dónde y cómo puede proteger la información mientras ésta va de su disco duro a otro que está en cualquier parte del mundo. Fíjese que cada nivel proporciona servicios útiles a los niveles precedente y sub-siguiente. Piense en esta organización estratificada como si fuera un puesto militar.

6.2.2. Seguridad a través de cortafuegos

Existen varios tipos de cortafuegos basados en el hardware. Cuando los paquetes de información llegan a una

red de trabajo, la mayoría los reciben usando un router de filtrado de paquetes. Este es uno de los tipos de cortafuegos. Un router de filtrado de paquetes es una aplicación de seguridad que deniega a alguien ajeno a la red el permiso de conectarse a aplicaciones que están en ella a menos que el router lo reconozca, basándose en su dirección IP. Los router más avanzados pueden, incluso, usar perfiles para identificar una llamada entrante. Un perfil incluye la dirección IP junto con información adicional sobre la llamada como el protocolo que está usando (FTP versus HTTP, por ejemplo) y las direcciones. Además de esto, algunas compañías utilizan dos router, basándose en el dicho cuanto más mejor. Este sistema de doble router se denomina *Bastion Host*.

El uso de un router puede evitar que los hacker que rompen las barreras de seguridad de bajo nivel utilicen aplicaciones de la red. Un nivel adicional de seguridad lo proporciona un servidor de autenticación que funciona con el router de filtrado. Este servidor funciona de un modo muy similar a otro tipo de tecnologías de autenticación: verifica que la persona que usa la dirección IP externa es realmente la persona a la que se le permite el uso de esa dirección. Juntos, un router de filtrado más un servidor de autenticación, componen el cortafuegos de uso más común actualmente en la seguridad de empresas.

***Nota:** Puede instalar un cortafuegos sólo de software. No tiene por qué ser hardware. Cada vez más se recomienda un cortafuegos de software a personas que navegan a través de conexiones de alta velocidad. Más adelante se describe algún software excelente que puede instalar en su ordenador.*

Otro tipo de cortafuegos hardware es el llamado servidor proxy, que puede chequear el contenido de la información de cada paquete que llega. Sin embargo, este tipo de chequeo puede ralentizar el sistema. Paradójicamente, cuando se usa con una memoria caché, un servidor proxy puede aumentar la velocidad de acceso a Internet. Puede actuar de la misma forma que una caché de páginas Web de un ordenador personal: cuando carga una página Web, la almacena en su disco duro. Después, si le pide al navegador que la busque otra vez, cosa que sucede muy a menudo, aparece rápidamente en pantalla la página almacenada. El acceso a los discos duros es más rápido que el acceso a través de Internet.

(Para ver su caché personal, abra a Internet Explorer, y vaya a Herramientas, Opciones de Internet y una vez allí, haga clic sobre el botón **Configuración** en la sección de **Archivos Temporales de Internet**, y escoja a continuación **Ver archivos**.)

Como puede imaginar, sitios de megaservidores como MSN y AOL aprovechan el servidor proxy para utilizarlo como caché. Cuando uno de sus clientes accede a la página principal de Yahoo, esta página permanece en el servidor proxy y es más fácilmente accesible para otros clientes que quieran acceder a esa página en los siguientes quince minutos. Las páginas cacheadas se purgan y se sustituyen regularmente porque los sitios Web más visitados cambian con frecuencia los títulos de las noticias y otros elementos.

Los servidores proxy tienen un uso adicional: pueden invertir la dirección habitual del filtrado de seguridad y ser instalado para evitar que los empleados tengan acceso a ciertas categorías de sitios de Internet.

***Nota:** Deberíamos mencionar también un popular protocolo (conjunto de normas) denominado RADIUS (Remote Authentication Dial in User Service, Autenticación de llamada remota durante el servicio del usuario). Puede ayudarle a organizar elementos de su seguridad y servir de asistente manteniendo listas de nombres de usuarios y contraseñas asociadas, junto con otros tipos de datos de seguridad. Cuando alguien ajeno intenta entrar, RADIUS les requiere la autenticación necesaria.*

6.3. Seguridad mediante cifrado

Aunque ponga todo su empeño, los cortafuegos son inútiles para evitar el robo de información durante el proceso de transmisión. Si un hacker puede grabar su información mientras ésta circula por Internet, ha conseguido ya romper su seguridad.

¿Qué debería hacer si un hacker consigue cruzar su cortafuegos por la fuerza bruta o por ingenuidad, o si ya está dentro? Recuerde que un empleado enfadado puede ser incluso más peligroso que los hacker externos. La respuesta es añadir información adicional cifrando información importante.

Si no puede evitar que entren a por sus archivos, puede cifrar los archivos hasta tal grado que el intruso no pueda nunca reconstruirlos (descifrarlos). El cifrado de información es, probablemente, el elemento de seguridad más fiable. Por este motivo, junto con el hecho de resulta fascinante para muchos de nosotros, más adelante en este libro nos dedicaremos a las diversas formas de cifrar información, sin llegar a hacerlo de tal forma que quede destruida totalmente y no se pueda recomponer nunca.

La información puede ser cifrada antes de ser almacenada en su disco duro o como parte del proceso de transmisión. Se habrá dado cuenta de que, cuando está a punto de dar el número de su tarjeta de crédito para comprar algo en Internet, se despliega un mensaje calmando sus miedos diciendo que la información está protegida por SSL (Secure Socket Layer). SSL es una forma de cifrar información antes de enviarla a través de Internet. Los cifrados SSL tienen una clave privada (sólo el emisor y el receptor la conocen). Hablaremos mucho más de claves más adelante.

Los peligros de las conexiones de alta velocidad

Habrás oído hablar de ello: un yuppie feliz acaba de contratar una conexión DSL (*Digital Subscriber Line*) o un módem por cable a su compañía telefónica. La velocidad de su conexión a Internet se dispara. Después de años viendo páginas Web con una lentitud de tortuga, ahora aparecen muchas a un tiempo. Puede ya ver imágenes de vídeo a toda velocidad. Las canciones que antes tardaba media hora en descargar llegan en menos de dos minutos. Es casi un nuevo aparato. Y, al contrario que la vieja conexión con el viejo módem pasado de moda, las líneas DSL y el módem vía cable siempre están conectados. No hay que marcar un número, sólo hacer clic sobre el navegador y ya está dentro.

Después de tener durante unos días una rápida conexión a Internet, el cliente feliz se acuerda del viejo dicho: no es oro todo lo que reluce. Cuando utiliza una conexión de este tipo, el ordenador queda abierto al mundo exterior de formas que no se esperan. Sin embargo, tranquilícese. No es que todos los hacker de Internet se den cuenta de que su puerta virtual está abierta y empiecen a merodear por su disco duro (robando archivos, dejando caballos de Troya o bombas lógicas). Me temo que usted no es tan importante.

Pero, en teoría, si deja activada la conexión en red o la opción de compartir archivos, es posible que entre un hacker. Por supuesto, puede apagar el ordenador cuando no lo está usando y evitar que pase algo mientras está fuera. Pero la mejor protección es desactivar la opción de compartir archivos.

Nota: Cada sistema operativo tiene su propia forma de desactivar la opción de compartir archivos. Por ejemplo...

*Windows 98 tiene que ir a Inicio, Configuración, Panel de control. Haga doble clic en el icono de Red. En la pestaña de Configuración, haga clic sobre el botón **Compartir archivos e impresoras**. Desactive la casilla que precede a la opción **Permitir que otros usuarios tengan acceso a mis archivos**. Haga clic sobre **Aceptar** dos veces para cerrar los cuadros de diálogo.*

¿Cuál es la velocidad de un módem por cable?

Prácticamente igual que otra conexión a Internet a la que pueda tener acceso hoy en día. El hardware propio del módem puede ir a 30Mbps (megabits por segundo). Pero, recuerde, ¿cuál es ese vínculo que patina en una cadena? En la cadena del módem por cable, el segundo punto débil es probable que sea la tarjeta de Internet del ordenador que se conecta al módem. Ethernet limita la transmisión a 10Mbps. El punto más flaco es, probablemente, el servidor del sitio Web al que está accediendo o, incluso, la propia estructura de Internet. Es probable que le permitan usar un 10 por ciento de la velocidad de Ethernet. Así que, por ahora, se está moviendo entre 5Mbps y 1Mbps. Pero no se ponga triste: 1Mbps es como la velocidad de un rayo si se compara con las tradicionales velocidades de acceso. La mayoría de las personas se conectan a Internet a la velocidad de 28.800Kbps (kilobits); compárelo con 1.048.576 (1Mbps).

7.1. ¿Qué hacemos?

¿Podría un hacker entrar en una máquina que tenga una conexión tradicional a Internet? Seguro. Pero con este tipo de conexión, cada vez que entra en Internet se le asigna una nueva dirección IP hasta que se desconecta del servidor o cierra el navegador. Es como si estuviese haciendo cada nueva llamada desde una cabina diferente; nadie podría identificar o recordar su número de conexión.

Pero si tiene una conexión IP estable y constante, su ordenador es completamente accesible. Sobre todo porque si detectan un puerto sin buenas defensas, pueden volver unos

días más tarde y explotarlo más porque su dirección no ha cambiado.

Hablaremos posteriormente de las posibilidades de que un hacker entre en su ordenador. Esto es tan llamativo para los medios de comunicación y tan escalofriante para nosotros, que llena los telediarios de la noche. Pero, poniendo los pies en la tierra, nunca se ha informado de un ataque de un hacker de esas características. Por supuesto, por muy ínfima que sea la posibilidad, todos nos sentimos como si se fuera a hacer público nuestro diario privado. A todos nos molesta que nos toquen la privacidad.

Por ejemplo, ¿sabe cuánta gente se negó en Estados Unidos a indicar en el censo sus ingresos anuales? ¿En qué estarían pensando? ¿No se dan cuenta de que hacen la declaración de la renta todos los años en el mes de abril? ¿Hasta qué punto es confidencial la información sobre el número de dormitorios que hay en su hogar?

La amenaza psicológica del hacker es superior a la real. Veamos qué es lo que sucede cuando se conecta a través de DSL, módem por cable o una LAN en el trabajo.

Con cualquiera de las tres opciones consigue una dirección IP invariable. Esto viene a ser el equivalente a tener el mismo número de teléfono durante años. Los hacker pueden rastrear Internet buscando direcciones, al igual que algunos televendedores rastrean listas de números de teléfono, buscando alguna alma cándida y cansada que esté tratando de tomarse su almuerzo.

Considere las posibilidades. Millones de máquinas están siempre conectadas a Internet, y probablemente entre ellas, unos miles de hacker fisgones. ¿Es probable que den con su diario? ¿Es este tan fantástico? Y, si así fuera, ¿cómo podrían saberlo? ¿Es una persona famosa cuya vida pública es muy conocida e interesante para otros?

De todas formas, cuando consigue una dirección IP permanente, un hacker podría probar su sistema y, tras haber reunido información sobre los problemas de sus defensas, recordar su dirección IP.

Esto les permite volver a entrar en el ordenador o compartir la dirección con otros. Por supuesto, si ha escrito un libro sobre los hacker, o hablado con ellos en *The Today Show*, probablemente habrá atraído la atención de más de uno, sin mencionar también a los whacker, cracker, slacker y algún que otro innombrable más. Algunos pueden no apreciar sus comentarios. Como autor de este libro, puede dar por seguro

que voy a ejecutar mi conexión por cable a través de un cortafuegos de ZoneAlarm antes de que lleguen a mi disco duro (más adelante se podrá información acerca de protección con cortafuegos personales).

Otra forma de aumentar las posibilidades que los hacker le ataquen es ir a donde ellos están. Por ejemplo, una buena forma de probar bien su sistema, es ir al grupo de noticias denominado alt.2600.hackerz, y hacer algún que otro comentario desagradable sobre la moralidad dentro de la comunidad hacker. A continuación, retírese y quédese observando como se lanzan como fieras.

También si tiene enemigos en el trabajo o en su pueblo, pueden hacerse hacker temporalmente, intentar entrar en su ordenador y... ¡quién sabe!

Recuerde que los hacker intercambian software entre ellos cuando lo encuentran necesario para sus quehaceres. Una utilidad de este tipo es el crawler que está constantemente analizando direcciones IP, probando las defensas de aquellas que estén abiertas o añadiéndolas a la lista de *pardillos de oro*.

Esta utilidad ping puede escanear miles de direcciones IP al mismo tiempo (el término ping se refiere al envío de una señal a una dirección IP para ver si está activa o abierta).

Puede descubrir que su conexión a Internet de banda ancha está siendo escaneada por este robot varias veces al día. Sin embargo, mientras haya desactivado la opción de compartir archivo, no pueden tener acceso a sus datos. No va a tener problemas, a no ser que sea uno de esos que ejecutan un servidor desde casa. Las empresas, por supuesto, casi siempre tienen un servidor y deberían tomar las precauciones necesarias descritas anteriormente.

7.2. Denegación de servicio

Alguna gente aún se preocupa de ser el objetivo de un ataque de "denegación de servicio", popularmente conocido como DOS, como aquel que bloqueó la CNN, Yahoo y otros sitios importantes a principios del 2000 y esporádicamente desde entonces. ¿Pero quiénes creerán que son estos agobiados? ¿Es acaso probable que un hacker gane puntos por ralentizar, o detener, su pequeño ordenador?. Si ni siquiera es un servidor, ¿verdad? Recuerde siempre que un hacker

quiere ganarse puntos, es decir, ganarse el respeto, generalmente de otros hacker. Es muy improbable que lo hagan denegando servicio a pequeños y desconocidos sistemas como el suyo.

7.3. ¿Puede convertirse en un *zombi*?

Hay una mayor probabilidad de que un hacker pueda tomar prestado su ordenador para comenzar un DOS. Normalmente, un DOS comienza cuando un hacker planta bombas lógicas en varios ordenadores de todo el mundo. Este uso de los ordenadores se ha llamado entre los hackers *zombis*.

Entonces, en el gran día, el hacker contacta con cada una de las bombas, da el pistoletazo de salida y comienza el ataque DOS. Se transmite una lluvia continua de intentos de acceso desde los *zombis* (la denegación de servicio es una sobrecarga masiva de solicitud de conexión, tan grande que los servidores no pueden hacer otra cosa que no sea atenderlos). Los intentos de descubrir a un hacker son mucho más difíciles porque los *zombis* inocentes son los que realmente están transmitiendo el ataque.

Pero de nuevo, no sobrestime su insignificante posición en el esquema de cosas. Los pequeños chicos de la "mafia de Canadá" conocidos como *mafia boys*, que según informes, fueron los autores del ataque de principios del 2000, usaron de *zombis* a máquinas muy poderosas y ultrarrápidas, como las de la Universidad de California de Santa Bárbara. ¿Cree que un hacker va a lanzar una transmisión de esas características desde su pequeño sistema, cuando hay tantos inquilinos alternativos poderosos?

Si teme el espionaje por parte del Gobierno, recuerde que esto no es Rumania. No se creará que hay tantos espías como para chequear las actividades de cada uno de nosotros, ¿verdad? Significaría que la mitad de la población de un país estaría observando a la otra mitad, cosa que, gracias a Dios, no sucede.

Por otra parte, debería recordar que el espionaje y formación de perfiles a través de Internet se hace la mayoría de las veces con ordenadores inagotables de alta velocidad que pueden observar su actividad en la red y cuánto tiempo pasa viendo los sitios que visita. Pueden construir su perfil sin intervención humana y cada ordenador de esos puede reali-

zar miles de perfiles al mismo tiempo. Ese peligro sí es real y puede pasar aquí.

Si tiene enemigos, secretos importantes, un sistema poderoso, un servidor, montones de dinero, un contrato que deba ser investigado por la CIA, la costumbre de burlarse de los hacker o algún otro motivo para temer un ataque, el siguiente capítulo le explica lo que puede hacer para proteger su sistema y ayudarle a dormir por las noches mientras su conexión continua está ahí reluciendo dentro de Internet como las bombillas del porche de su casa.

Cómo proteger su banda ancha

Después de tanto esperar, por fin ha conseguido que la compañía de teléfonos le instale su conexión DSL o su módem por cable. Ahora los sitios Web aparecen inmediatamente en su pantalla con tan sólo hacer clic en el vínculo. ¡Por fin ha conseguido su conexión de banda ancha! (También conocida como banda de alta velocidad y simplemente significa que ha conseguido un ancho de banda mayor que el que tenía antes. Compárelo con un grifo de ducha: entra mucha más agua ahora que cuando era más estrecho).

Además de la alta velocidad de acceso, también obtiene una dirección IP permanente y estática, y, si no hace bien las cosas, los hacker lo tienen ahora más fácil para entrar en su ordenador y copiar o borrar archivos.

8.1. Primero, seguridad

Varios proveedores de banda ancha están actualmente reforzando sus avisos de seguridad y características, aunque la mayoría están de acuerdo en que la amenaza se exagera mucho en los medios de comunicación (recuerde que todavía no se sabe de ningún caso de alguien que haya sufrido un ataque de hacker vía cable o conexión DSL). EarthLink, por ejemplo, espera ofrecer un servicio de cortafuegos junto con la conexión DSL a finales del 2000. *Excite@Home* proporciona opcionalmente utilidades de antivirus y cortafuegos de *McAfee.com*. Es posible que los proveedores de banda ancha comiencen a incluir el software del cortafuegos con el hardware de conexión que ofrecen.

Sin embargo, por ahora, no hay motivo para que nosotros, usuarios normales, nos preocupemos al usar ordenadores personales en casa. Pero los hacker fuerzan continuamente los límites de lo posible, así que el mes que viene puede ser todo diferente.

8.1.1. Cómo atraer a los hacker

Desactivar la opción de compartir archivos en su ordenador es el único paso esencial que tiene que dar para bloquear el acceso a los hacker. Anteriormente se ha explicado cómo y, posteriormente en este capítulo, verá cómo hacerlo. Sin embargo, debería preocuparse si alguno de los siguientes aspectos se corresponde con su descripción:

1. Le ha faltado al respecto a los hacker públicamente, por ejemplo, participando en el grupo de noticias alt.2600 para insultarlos, o escribir un libro sobre hacker que los describe como "moralmente inmaduros".
2. Tiene otros enemigos que son especialistas en tecnología.
3. Tiene mucho dinero o secretos importantes.
4. Controla un sistema poderoso.
5. Controla un servidor.
6. Ha sufrido ya ataques de hacker.
7. Controla una empresa u organización, especialmente aquellas que están relacionadas con gobiernos, militares, grupos políticos u otras actividades de interés para los hacker.

8.2. Instalar un ZoneAlarm

Gratis para usuarios particulares y a 4.000 Ptas. por sitio para empresas, administración o centros educativos, ZoneAlarm (www.zone1abs.com) es una solución común para aquellos usuarios preocupados por los posibles ataques de los hacker. Funciona con Windows 95, 98, NT 4.0 o Windows 2000 (si ha probado la versión Beta, use sólo la última versión disponible de Windows 2000).

De todos los cortafuegos personales que he visto, este es actualmente el mejor. Es resistente, sólido, muy efectivo a la hora de proteger su ordenador, muy bien desarrollado y fácil de instalar y usar, instructivo y duro como una pared de

acero de tres metros de grosor. ActíVELO y relájese. Es mi consejo.

Además, ZoneAlarm tiene el precio ideal: cero pesetas para usuarios particulares.

ZoneAlarm es un cortafuegos basado en software que puede interponerse firmemente entre su vulnerable y traqueteante disco duro y la grande y mala World Wide Web y las arañas horribles y peludas que se mueven a través de ella.

Como ya sabe, si ha leído este libro desde el principio, los hacker, cracker, whacker y sus malvados robot están buscando continuamente a través de la red conexiones permanentes de banda ancha.

Si tiene información que no le gustaría compartir con extraños, o le preocupa que un intruso pueda acabar con sus archivos, considere la posibilidad de descargar ZoneAlarm. Se caracteriza por tener varios niveles de seguridad y también le permite definir qué aplicaciones están autorizadas a acceder a Internet. Además de los navegadores y los programas de gestión de correo electrónico, otros programas como Word o RealPlayer pueden conectarse a la red. Un cortafuegos no sólo controla los intentos de entrada en el sistema a través de Internet, sino también los de salida.

Si una aplicación intenta acceder a Internet, ZoneAlarm le avisa de ello. En este momento, puede permitir, rechazar o establecer una norma para el futuro, de forma que no tenga que responder cada vez que el software realiza el intento.

8.2.1. Enciérrelos por completo

Una de las mejores características de ZoneAlarm es que le puede ordenar una total desconexión de Internet. Si se está desarrollando un ataque, tecleando Ctrl-S se bloquea toda la comunicación con Internet. También puede activar el sistema de desconexión después de cierto tiempo de inactividad (de la misma forma que salta el protector de pantalla o se apaga el monitor cuando ha estado sin tocar el teclado o el ratón durante un tiempo). De esta forma, se desconecta la conexión de banda ancha.

Pero incluso mientras está navegando por Internet o enviando un correo electrónico, ZoneAlarm le protege de forma muy efectiva, porque las arañas de rastreo u otros sistemas de escaneado no encontrarán nada más que invisibilidad y silencio. Esos malvados *ping* llamando a sus

puertas virtuales no verán ni oirán nada. Es como si su ordenador no estuviese conectado o ni siquiera existiese.

Siga estos pasos cuando instale ZoneAlarm:

1. Teclee esta dirección en el navegador:
hotfiles.zdnet.com/cgi-bin/texis/swlib/hotfiles/info.html?code=00157&=1od

Si ya no se encuentra disponible ahí, vaya a:

www.zdnet.com/special/filters/defense/action o www.zdnet.com y localice **ZoneAlarm**.

También puede localizar a través de un motor de búsqueda como Google **Shields Up!**, la página principal de ZoneAlarm.

2. Haga clic sobre el botón de **Download** (descarga del programa).
3. Cuando se le pregunte, elija **Save the file on hard disk** (guardar el archivo en su disco duro).
4. Localice el archivo **Zonalarm.exe** en su disco duro y haga clic sobre él para instalar el programa.
5. Haga clic sobre el botón **Next** (siguiente) durante el proceso de instalación.

Cuando se complete la instalación aparecerá ZoneAlarm, tal y como se muestra en la figura 8.1.



Figura 8.1. Este es el panel de control de Zone Alarm

Si lo desea, ZoneAlarm puede ejecutarse automáticamente cada vez que encienda su ordenador. Como la mayoría del buen software, esta opción y muchas otras se puede cambiar para adecuarse a sus preferencias. Fíjese que todos los profesionales del mundo de la informática y mucha gente normal dejan sus ordenadores encendidos todo el rato para evitar cambios bruscos de temperatura al arrancar el ordenador. Es un método preventivo contra el mal funcionamiento prematuro de chips y demás hardware.

***Nota:** Si quiere información adicional sobre ZoneAlarm, visite su sitio Web: www.zonelabs.com/support.htm.*

Desde que he comenzado a usar ZoneAlarm con mi conexión por cable, me he encontrado con una media de diez intentos de entrada diarios en mi ordenador. Sin embargo, esta actividad aumenta rápidamente cuando visita ciertos sitios. Por ejemplo, sitios como Napster, que permite compartir archivos, incrementan muchísimo estas peticiones de acceso.

Estos intentos de entrada son efectuados por una persona o su agente robotizado. Cuando se producen, ZoneAlarm, opcionalmente, despliega una alerta del siguiente modo:

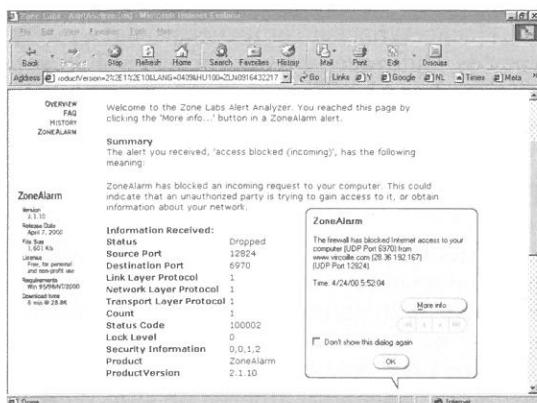


Figura 8.2. Haga clic sobre el botón **More Info** si alguien intenta entrar en su ordenador y ZoneAlarm le llevará a este sitio Web (en la figura, detrás de la ventana de alerta de ZoneAlarm)

8.3. Otros cortafuegos personales

Si es de los que le gusta mirar diferentes cosas antes de comprar, quizá le interese probar con otros programas cortafuegos para uso personal o de empresa. Symantec, por ejemplo, ofrece Norton Internet Security 2000, que incorpora el cortafuegos a su programa antivirus. Otro también muy conocido es **BlackICE Defender** de Network ICE. El término

ice (hielo) se usa a veces para describir las defensas de los ordenadores y fue creado por William Gibson en sus novelas cyberpunk. ICE significa *Intrusion Countermeasure Electronics* (sistema electrónico para combatir la intrusión).

ZoneAlarm, la utilidades **Norton** y **BlackICE Defender** están disponibles en el CD que acompaña al libro.

8.4. Hagamos una prueba

Los cortafuegos son fantásticos y, si quiere ver si necesita uno, existe software que imita a un hacker intentando entrar en su ordenador. Pruebe con **HackerWhacker**. Ofrece un escaneo gratuito que le muestra qué es lo que puede llegar a su ordenador. Es realmente escalofriante que alguien intente entrar en su ordenador. **HackerWhacker** comprueba si tiene puertos TCP o UDP expuestos, algún acceso a través de NetBIOS, si tiene activada la opción de archivo compartido (¡lo mato!) y chequea ciertos tipos de riesgos del servidor Web CGI.

8.4.1. Cuidado con los PWS

Las verificaciones de CGI realizadas por **HackerWhacker** deberían ser necesarias sólo para aquellas personas que tengan un servidor Web activo. Sin embargo, podría sorprenderse al saber que podría estar exponiéndose a sí mismo como "servidor Web" a los internautas sin ni siquiera darse cuenta de que su ordenador alberga un sitio Web. Windows incluye una fantástica característica llamada Servidor Web Personal PWS. Es un tipo de software que simplifica el diseño y chequeo de un sitio Web. Si usa FrontPage o Visual InterDev para diseñar un sitio Web, puede usar el PWS para ver cómo quedaría y funcionaría sin tener que llegar a cargarlo en el hardware del servidor de Internet. De esta forma, puede realizar todos sus diseños desde su ordenador personal. Sin embargo, PWS podría dejarse conectado cuando no está diseñando la página y PWS puede permitir conexiones desde el exterior, como un servidor Web. Los test del CGI de **HackerWhacker** le avisarán si este dispositivo está conectado y siendo demasiado amable con extraños.

HackerWhacker es completo e informativo. Para iniciar su escaneo de seguridad, debe desconectar primero **Zone Alarm**, si lo está usando, haciendo clic con el botón derecho

del ratón sobre el icono de ZoneAlarm en la barra de tareas y escogiendo la opción **Shutdown**.

8.4.2. ¿Hay extraños en su ordenador?

Vaya a www.hackerwhacker.com. Verá esta alarmante e inteligente pregunta: ¿hay extraños en su ordenador? Verá montones de hipervínculos, herramientas e información interesante sobre los hacker y temas de seguridad, tal y como se muestra en la figura 8.3

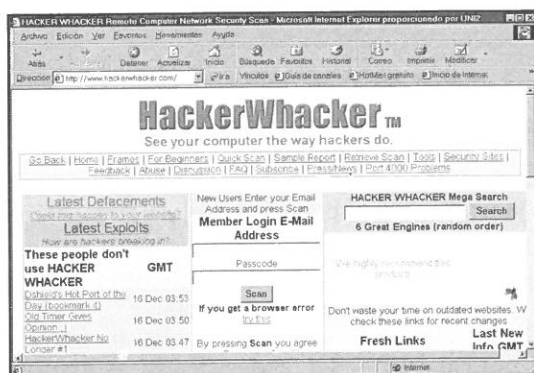


Figura 8.3. Este es un sitio excelente para la gente interesada en diversos aspectos de ciber-seguridad

Haga clic sobre el link **Free Security Scan**. Se le pedirá su dirección de correo electrónico y le enviarán a través de él las instrucciones. Le asignarán una dirección de Internet a la que ir y se le ofrecerán distintos tipos de test.

***Nota:** Si su ordenador forma parte de la red de una empresa, consulte con alguien del departamento de informática antes de realizar un escaneo de seguridad.*

Verá distintos test que le mostrarán la información y, posiblemente, los archivos de su sistema que están disponibles. También verá los puertos que tiene abiertos (si es que tiene alguno) y muchos vínculos y sugerencias sol

subsanan vulnerabilidades en sus defensas. La figura 8.4 le muestra una típica hoja de resultados

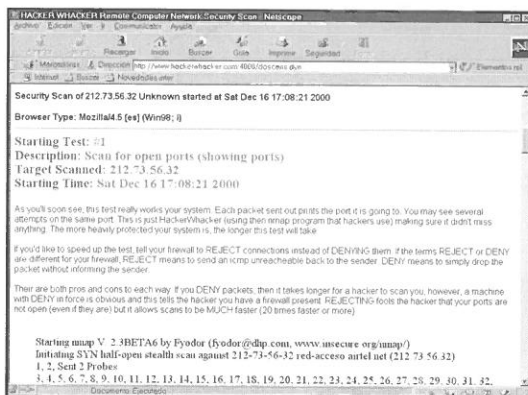


Figura 8.4. HackerWhacker chequeó un número de puertos y vio que tenía el puerto 139 totalmente abierto

HackerWhacker encontró un puerto abierto en mi sistema y me dio las siguientes sugerencias:

"Un OPEN significa que su ordenador aceptará conexiones en el puerto TCP/UDP especificado. A menos que entienda por qué este puerto TCP/UDP está abierto y quiere que sea así, esto está mal (BAD). Tiene que entender para qué son usados sus puertos TCP/UDP (vea vínculos abajo), porque puede preferir que ciertos programas de su ordenador queden fuera de acceso." Siga los hipervínculos proporcionados por la página de información de HackerWhacker y verá soluciones a cualquiera de los problemas que le aparezcan en el informe de HackerWhacker.

HackerWhacker también ofrece servicios adicionales, así que visite su página si quiere ver otro tipo de pruebas y asistencia sobre seguridad.

***Nota:** El término puerto se refiere a cada una de las 65.535 direcciones potenciales disponibles en una máquina que esté ejecutando software TCP/IP. Al igual que sucede con la mayoría de la programación informática, han surgido ciertas convenciones referentes a la asignación de puertos a*

funciones específicas. Por ejemplo, el puerto 23 es Telnet, el puerto 25 es el de transferencia de correo electrónico (SMTP) y el puerto 139 es el de servicio de sesión de NetBIOS.

8.4.3. Pruebe con el escaneado gratuito de Symantec

Puede encontrar otros escáneres de seguridad gratuitos en Internet. Pruebe este sitio de seguridad ZDNet en www.zdnet.com/zdtv/cybercrime. En el momento de escribirse este libro, había un hipervínculo en esta página a un escaneo gratuito de Symantec, que comprueba su cortafuegos, informa si ciertos contenidos para adultos se pueden descargar en su sistema, si tiene una protección antivirus Shields Up! y si alguien puede obtener información personal suya, como por ejemplo, el último sitio visitado y su dirección de correo electrónico.

8.5. Honeypots y otras tácticas

Si descubre que un puerto está inexplicablemente abierto en su sistema, podría significar que un caballo de Troya lo está manteniendo así para una posterior incursión por parte del hacker que lo ha puesto ahí.

Si tiene problemas con los hacker, es posible que intente frenarlos poniendo *honeypots*. Puede bajar un buen *honeypot* llamado DTK (Deception Tool Kit) en la página Web www.all.net/dtk.

La idea es que atraiga a los hacker al igual que esas tiras de papel lo hacen con las moscas. El *honeypot* finge ofrecer entradas, exponer tentadores servicios de red en los puertos de su ordenador esperando que un hacker venga a aprovecharse de ellos. En realidad, se registran y bloquean los movimientos del hacker sin dejar de jugar con él. El hacker piensa que está dentro de un sistema seguro, agradable, pero lo que en realidad sucede es que se está delatando mientras se registran sus movimientos.

8.5.1. Pruebe con Shields Up!

Steve Gibson nos hace un gran favor a todos con este sitio Web llamado *Shields Up!* Échele un vistazo a <http://www.shieldsup.com>

grc.com/x/ne.dll?bh0bkyd2 o busque *Shields Up!* usando su motor de búsqueda favorito. Una de las mejores cualidades de Gibson son sus claras y detalladas explicaciones de la complicada y confusa seguridad de Windows, Internet y características de trabajo en red. Explica cómo puede saber si todo funciona, lo que puede hacer para incrementar la seguridad de su sistema y no obvia los aspectos más complicados. Le ayudará a entenderlos. Si es un administrador que está al cargo de la seguridad de un sistema que se base en Microsoft, o alguien que simplemente quiera profundizar más en estos temas de seguridad en Internet, visite esta página y leala. Incluso si es un usuario normal, también le recomendaría que le echase un vistazo. Es tiempo bien empleado.

8.6. No son precisamente las "cookies" danesas

Uno de los puntos débiles de la seguridad son las cookies, una característica que pretende facilitarle la vida. Tiene que haber una forma de guardar sus preferencias. Muchos sitios Web le permiten configurar sus páginas a su gusto, por ejemplo, definir los colores que quiere ver, qué tipos de nuevas categorías aparezcan, los contenidos de la carta de compras y otros datos. Pero, ¿qué hacen para recordar estas elecciones? Cuando las vuelve a visitar, ¿cómo hacen para saber que es uno mismo y decir "Bienvenido, Richard"?

La respuesta es que recogen esa información de su disco duro. Si, hay comunicación entre los sitios Web de Internet y su disco duro. Y esto es sólo la punta del iceberg; algunos sitios recogen información sobre el usuario cuando visita sus páginas (qué hipervínculos escoge seguir, qué compra y otros patrones de comportamiento personal). Alguna de esta información se recoge activamente mientras navega, pero otra, como la personalización de sitios, se guarda en las denominadas "cookies". Los archivos de cookies se almacenan en su disco duro y pueden ser leídos posteriormente por los sitios que los guardaron o por otros. Esto puede ser una ventaja; por ejemplo, las cookies pueden guardar contraseñas y nombres de usuario de forma que no tenga que escribirlos cada vez que visita un sitio que requiere esta información. Pero las cookies pueden proporcionarles a otros

información sobre sus movimientos en la red y más cosas que seguramente no quiere que se sepan. Las cookies pueden describir las fotos que ha estado mirando, cuánto tiempo ha estado mirando cada una de ellas y, consecuentemente, formar su perfil personal.

Aquí tiene un experimento que puede probar:

1. Ejecute Internet Explorer.
2. Elija la opción Herramientas, Opciones de Internet.
3. Haga clic sobre la pestaña de **Seguridad**.
4. Haga clic sobre **Personalizar Nivel**.
5. Baje por la lista de opciones hasta que vea **Cookies**.
6. Haga clic en la opción **Pedir datos en Permitir que se almacenen cookies en el equipo**. Haga clic dos veces en **Aceptar** para cerrar las opciones de los cuadros de diálogo.

Ahora vaya a visitar cualquiera de sus páginas Web favoritas. Se sorprenderá de la cantidad de veces que le aparecerá el cuadro de diálogo sobre cookies, tal y como se muestra en la figura 8.5

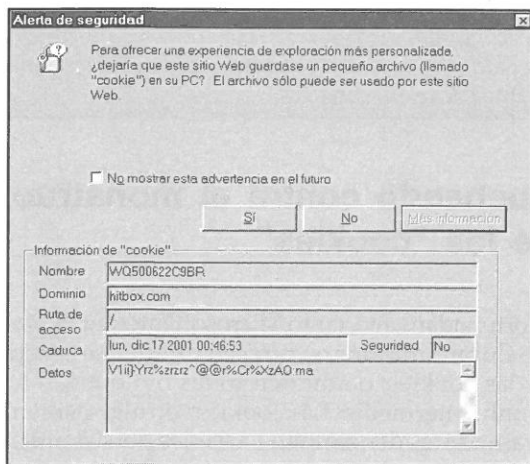


Figura 8.5. Este pequeño cuadro de diálogo aparecerá docenas de veces mientras navegue por Internet, si escoge la opción de ser avisado cada vez que una cookie esté a punto de ser guardada en su disco duro.

La alerta de cookie que se muestra en la figura 8.5 es muy interesante. Estaba mirando un famoso sitio Web de bolsa (sólo noticias y una página sobre finanzas). Sin embargo, la cookie que se mostró estaba disponible para otro sitio completamente diferente. De hecho, la cookie (he alterado la información que se ve en la figura 8.5 para proteger al culpable) almacenó datos sobre mi interés en determinada información financiera. A esta información podía acceder algún famoso sitio de bolsa en Internet.

Otro experimento que puede intentar es el siguiente: eche un vistazo en una carpeta llamada Cookies en el directorio de Windows. Lea cualquiera de los archivos .TXT (cada uno es una cookie) haciendo doble clic sobre cualquiera de ellos. Fíjese que el sitio relacionado con esa cookie aparece entre los datos como una dirección típica de Internet: www.alguna parte.com. Mi carpeta de cookies tiene más de 500.000 bytes de datos y casi mil cookies. ¿Y la suya?

Se pueden leer algunas cookies porque recogen datos del estilo: *Date Visited 9-9-00/Time Visited ...* etc. Otras, sin embargo, están cifradas y sólo puede leerlas el software del sitio Web que las guardó.

***Nota:** Netscape no almacena cada cookie en un archivo separado como hace Explorer. Usa un único gran archivo denominado cookies.txt y guarda cada cookie como una única línea de ese archivo.*

8.7. Luchando contra el monstruo de las "cookies"

Desafortunadamente, tanto Microsoft Internet Explorer como Netscape Communicator no ofrecen utilidades prácticas para bloquear las cookies; o admiten todas o bloquean todas. No hay un punto intermedio. Las cookies son algo así como lo que los programadores denominan variables persistentes. Pueden almacenar la información de sus visitas en la red. Es una buena característica. Sin embargo, también ayudan a los almacenes de datos a construir lo que llaman modelos predictivos (predicen sus futuros movimientos y sus compras).

Actualmente, Microsoft está planeando incluir una característica en Internet Explorer (IE) versión 5.5 que pueda

distinguir las cookies de almacenamiento de datos acerca de nuestro perfil de las cookies inofensivas y útiles, como las que recuerdan las contraseñas. Sin embargo, empresas de publicidad, marketing y comercio electrónico están protestando por esta nueva característica del IE que daría a los usuarios una alternativa al actual bloqueo total/sin bloqueo de cookies. El IE 5.5 puede distinguir entre cookies a la primera parte (aquellas que sólo devuelven información al sitio Web que las dejó ahí) o a terceras partes (aquellas que se envían a otros sitios). Si activa esta opción, cada vez que una cookie a terceras partes intente instalarse en el ordenador, se abrirá un cuadro de diálogo preguntando si quiere permitir el almacenamiento. Dado el número de cookies que llegan a un navegador Web típico, este cuadro de diálogo podría llegar a ser muy pesado. Quizás se desarrolle un enfoque más eficiente cuando se haya finalizado el producto.

Si quiere que sólo se almacenen cookies útiles en su disco duro y bloquear el resto, existen ciertas utilidades comerciales que pueden ayudarle en este trabajo de filtrado. Por ejemplo Norton Internet Security, McAfee Software's Guard Dog, Luckman Interactive (www.cookiecentral.com) y interMute (www.intermute.com).

Privacidad en Internet

Hasta este punto, este libro se ha centrado sobre todo en las distintas formas en las que los hacker entran en los sistemas informáticos para dañar la información. Sin embargo, hay una segunda amenaza importante en Internet: la acumulación de información que viola su privacidad.

La mayoría de la gente piensa que, si utiliza un alias al participar en un grupo de noticias, enviar correo electrónico o chatear, mantiene el anonimato. Creen que nadie sabe que, por error, han llegado a una de esas páginas calientes o que han comprado un CD *Eminem*. Asumen que su dirección de correo electrónico, sus hábitos de compra u otra información es totalmente confidencial.

¿No quedaría horrorizado si levantase la vista, viese un agujerito en el techo y de repente un ojo húmedo mirando a través de él? Pues alguien está en el ático haciendo una lista de todas las páginas Web que visita, de sus compras en Internet, de los mensajes que envía... todo lo que haga en la red.

Bien, pues empiece ya a sentir ese grado de espanto porque su ordenador y sus movimientos en Internet están sometidos a ese nivel de inspección.

Afortunadamente, existen buenos programas que puede empezar a usar y que le harán permanecer anónimo mientras navega por Internet como si fuese el hombre invisible. Posteriormente, le contaré cómo en la sección titulada "Contraataquemos".

*Nota: Si no cree que revele información sobre sí mismo mientras navega por Internet, vaya a www.privacy.net y haga clic sobre el botón **Full Analysis** (análisis completo)*

El informe tardará uno o dos minutos, pero le abrirá los ojos. Y, además, este informe sólo despliega la punta del iceberg de todo el espionaje de datos que se está llevando a cabo.

9.1. Ciberespionaje

Existen varios tipos de violaciones de privacidad en Internet: los que observan qué teclas presiona, los monstruos de las "cookies" y otras técnicas que siguen sus pasos y almacenan lo que ha hecho con su ordenador. Maridos a punto de divorciarse, jefes crueles y otros pueden tener razones para rastrear sus movimientos (un jefe puede incluso querer saber cuántas teclas presiona cada hora su empleado y cómo utiliza Internet mientras trabaja).

Una variedad de ciberespionaje lo realizan las empresas que observan y graban los sitios que visita, cuántos artículos lee, qué compras realiza, etc. A continuación construye un perfil personal sobre su comportamiento en la red que se puede vender como si fuese una precisa descripción de su personalidad, de sus necesidades (todas sus necesidades) y detalles sobre su situación económica, incluido su número de Visa.

No crea que cuando navega está solo en la habitación con la puerta cerrada; hay otros mirando y recordando sus movimientos con una mayor precisión que un mirón desde el techo de su casa. Se pueden adjuntar pequeñas etiquetas a las páginas Web, que son las que informan de su visita a un sitio Web y como se movió mientras estuvo allí.

Se tienen datos de que empresas como DoubleClick, AvenueA y MatchLogic han recolectado datos sobre hábitos de navegación con fines de marketing y publicitarios. Pueden añadirse unas pequeñas etiquetas invisibles a las páginas Web, que son las que informan de su visita a un sitio Web y lo que hizo en él.



Nota: Si no le gusta la idea de que otros sepan que ha buscado un libro concreto o que tiene interés por la jardinería, puede probar la utilidad de Idcide (www.idcide.com). Denominada Privacy Companion (la Empresa de la Priva-

cidad), la utilidad le muestra cuándo y cómo se siguen sus movimientos en Internet. Este navegador add-on distingue entre cookies útiles (que personalizan sitios) y cookies molestas que pueden recoger y transmitir información acerca de usted. La Privacy Companion le avisa y entonces decide qué hacer con esas cookies.

9.1.1. Herramientas del oficio

Las empresas pueden ahorrar dinero si enfocan bien su publicidad. Es obvio que obtener información sobre sus hábitos es muy útil para ellos. ¿Nunca se ha preguntado de dónde salen ciertos tipos de correo electrónico basura? ¿No cree que alguno de esos mensajes que le aparecen en la bandeja de entrada reflejan sus intereses increíblemente bien?

¿No ha usado nunca Internet para buscar, por ejemplo, unas posibles vacaciones en Las Bermudas y recibir de repente un par de folletos electrónicos sobre viajar a países exóticos?

A menos que le haya proporcionado a una agencia de viajes en Internet su dirección de correo electrónico, es probable que fuese observado mientras buscaba la palabra Bermudas.

9.1.1.1. Múltiples versiones de su persona

La información es poder y, por lo tanto, comercializable. Después de que haya navegado por Internet durante unos meses, muchas bases de datos tendrán pequeñas versiones de su persona: su nombre, dirección, teléfono, números de tarjeta de crédito y el segundo apellido de sus padres. Toda esta información proviene de formularios que ha rellenado al hacer alguna compra a través de Internet o al "registrarse", como lo llaman algunos sitios. Otra información sobre lo que le gusta o no, proviene de programas y cookies que pueden saber qué grupos de noticias y sitios Web ha visitado o los hipervínculos que ha seleccionado. Lo que es aún peor, algunos de los propietarios de esas bases de datos se juntan y agrupan toda la información, construyendo una versiones aún más detalladas acerca de su persona, que existen en el ciberespacio. Crean una escultura virtual, modificándola constantemente cada vez que se conecta a Internet y

navega, de forma que aumente la similitud cada vez más. Les está ayudando a tallar un simulacro eléctrico, y mucha gente piensa que esto no está nada bien.

Es tan espectral que se podría decir que están haciendo una réplica de uno mismo. Pero no olvide que hay otros problemas. Por ejemplo, algunos de sus datos pueden ser completamente falsos. Pueden introducirse errores en su retrato virtual que podrían perjudicarle el resto de su vida. Y, al contrario que los errores cometidos contra su integridad personal, no hay caminos legales o prácticos para enmendar falsos datos.

Por ejemplo, supongamos que visita una página en la que se habla de cómo robar un coche. Quizás le haya llegado por casualidad o por pura curiosidad. No importa, ha quedado ya registrado en su perfil personal. Esta información puede, y ha sido utilizada en vistas judiciales sobre custodias, juicios de divorcio y otros casos legales. Un hombre homosexual no fue admitido en la marina como resultado de la información que obtuvieron de él en Internet. Puede ser que, misteriosamente, le cueste contratar un seguro o encontrar trabajo. Hay usos múltiples de una ciber réplica y la mayoría de ellos son bastante desagradables.

En los Estados Unidos, el número de autorizaciones judiciales de búsqueda y de citaciones está aumentando a gran velocidad en todos los proveedores de servicio de Internet. Por ejemplo, en AOL sólo se presentaron 33 órdenes judiciales en 1997, 301 en 1999 y 191 en la primera mitad del año 2000. La mayoría de estas órdenes no exigen únicamente la presentación de todo su correo electrónico, sino toda la información que el Servidor tiene sobre uno: desde dónde se conectó, cuándo se conectó y durante cuánto tiempo, qué hizo en la red, qué sitios Web visitó, en qué chats participó y qué dijo.

En un esfuerzo por autocontrolarse, un grupo de empresas de publicidad y recolección de información (un 90 por ciento de las que pertenecen a este sector) propuso un plan en agosto de 2000 que, entre otras cosas, nos darían a los navegantes los siguientes derechos:

1. Tendremos un "acceso razonable" a nuestros perfiles (la información que tienen sobre nosotros).
2. No identificarán personalmente ningún dato que obtengan relativo a información económica importante, sexual o de salud.
3. No recogerán datos sobre nuestra Seguridad Social.

4. No informarán a ninguna persona física de ningún perfil que se esté formando en cada sitio Web y permitirá a las personas físicas rechazar esta colección de datos (cómo va a funcionar esto todavía no ha sido descrito. Es difícil imaginar cómo podría hacerse sin molestar al navegante).

Este es claramente un paso en la dirección adecuada pero la autorregulación no siempre funciona. Y hay un 10 por ciento de las empresas que no han aceptado el plan.

9.1.1.2. Cuando van cuesta abajo

Algunas compañías .com han dado en quiebra y parece que no se han preocupado de sus políticas de privacidad durante el cierre. Aparentemente, sería un buen momento para vender la información acerca de sus clientes y pagar algunas deudas.

Por ejemplo, Toysmart.com ha cerrado recientemente. Se dice que poco después del cierre, la empresa empezó a anunciar la venta de la base de datos de sus clientes, aún cuando su cláusula sobre privacidad dice claramente que tal información "no se comparte nunca con una tercera parte".

9.1.1.3. Legislación futura

En Estados Unidos, la Comisión Federal de Comercio y el Congreso están trabajando en legislar la protección de la privacidad en la red de varias formas: estándares de firmas digitales, requisito de que todo correo electrónico lleve una dirección de respuesta exacta (¡cuidadito los que enviáis correo basura!) reglas de establecimiento de privacidad y otras salvaguardas sobre la información que circula por Internet.

La mayoría de los sitios Web incluyen actualmente una cláusula sobre política de privacidad en alguna parte. Pero la mayoría de nosotros cuando navegamos, no nos molestamos en leer toda esa parafernalia. Creo que el porcentaje de personas que leen esos documentos que vienen con el software es de uno por cada veinte mil.

Cada vez es más obvio que las normas de privacidad deberían aplicarse uniformemente a todos los sitios Web, de la misma forma que hay normas que todas las compañías de teléfono deben seguir con respecto a sus llamadas y la información que intercambia durante esas llamadas.

La máquina Carnivore

¿Quién habría pensado que el FBI conseguiría permiso de los proveedores de servicios de Internet para instalar una máquina que controla el correo electrónico y otras comunicaciones a través de la red? Esta caja, llamada Carnivore, está equipada con muchísimo software. Se adjunta a un proveedor de servicio de Internet (ISP) donde puede "rastrear" (leer y controlar) las comunicaciones que fluyen a través de ese proveedor (en otras palabras, todo).

Está claro que el que bautizó a la máquina como Carnivore no pertenecía al departamento de relaciones públicas del FBI. Qué espeluznante nombre, aunque, la verdad, muy descriptivo.

Instalado en varios ISP en marzo de 2000, la existencia de Carnivore sólo salió a la luz pública cuando a finales de julio EarthLink, uno de los mayores ISP, se negó a instalarlo, alegando que le preocupaba la privacidad así como los fallos en el sistema.

Se dispone de pocos detalles porque en la actualidad EarthLink está implicada en un proceso judicial intentando impedir que el FBI le fuerce a instalarlo. Otra razón de que se sepa poco es que estamos hablando del FBI.

La Fiscal General Janet Reno dijo que iba a revisar el sistema y a asegurarse de que el FBI lo estaba usando "de forma coherente y equilibrada". El FBI dice que Carnivore no interfiere con la estabilidad ni impacta la ejecución de un sistema ISP. También afirman que tiene la habilidad de distinguir entre el tráfico general, al que ignora, y las comunicaciones que puede interceptar legalmente. Insisten en que Carnivore sólo graba información relacionada con las investigaciones del FBI. Por supuesto, de esto podría deducirse que descubre información que da lugar a nuevas investigaciones. Podemos cuestionarnos lo siguiente: si están llevando a cabo una investigación, ¿no sería más fácil intervenir las comunicaciones del individuo en concreto en su origen (casa, oficina, teléfono) como se ha hecho siempre? ¿Por qué esa necesidad de controlar las actividades de todas las personas que navegan por Internet, en vez de controlar

un traficante de droga aquí, un terrorista allí y un hacker allá?

Con el permiso de los jueces, Carnivore ha estado husmeando el correo electrónico de los más grandes proveedores de Internet desde marzo de 2000. El FBI dice que lo ha empleado en unos diez casos al mes. El Congreso interrogó a los oficiales del Servicio Secreto, aunque se hicieron públicos muy pocos detalles. El FBI dió permiso para que dos profesores universitarios independientes examinaran el software para verificar que Carnivore utiliza potentes filtros para observar y almacenar correo electrónico importante.

Un problema que preocupa a los defensores del derecho a la privacidad es la diferencia entre la tradicional intervención telefónica y el husmeo entre correo electrónico. Los métodos de investigación tradicionales, como leer el correo, buscar en la basura o grabar llamadas telefónicas son muy ineficaces. Los agentes tienen que hacer todo este trabajo en tiempo real, así que es imposible investigar a grandes masas de gente. Pero la investigación informatizada es algo completamente diferente. El correo electrónico y toda la información que fluye por la red son electrónicos, por tanto, pueden ser analizados electrónicamente. Esto significa que un sabueso como Carnivore puede almacenar información de todas las comunicaciones, y no sólo de unas cuantas. Seguro que está pensando cómo podrían almacenar toda esa información. Mire las estadísticas. Todos hemos oído hablar de que los ordenadores son cada vez más potentes, pero pocos consideramos que este incremento exponencial también afecta al coste del almacenamiento de datos. El año pasado un disco comprimido de 100 MB costaba 1.850 pesetas; este año, un CD-R de 650 MB cuesta 95 pesetas. Fíjese que, aunque la capacidad de almacenamiento de datos se incrementa rápidamente, la información que almacenan no aumenta en sí misma, es decir, la longitud de una palabra o su número de la Seguridad Social siguen siendo iguales. Y aunque aumente, por ejemplo, la cantidad de correo electrónico que envía, esta lo hace de forma gradual y no exponencial. La extensión de una película, de una sinfonía o la cantidad de información que una persona genera en su vida tienen un límite finito. Pero parece no haber límite para la capacidad de los

soportes de almacenamiento de datos, que aumenta rápidamente.

Pongamos por caso que su media de información en correo electrónico es de 2 Mb anuales, incluidos los mensajes de correo basura. Si lo trasladamos a costes reales de almacenamiento, todo el correo electrónico de una persona a lo largo de su vida hasta la edad de 70 años podría guardarse en una quinta parte de un CD-ROM, que hoy están a 95 pesetas la unidad. Vemos que el coste de almacenamiento de sus datos de correo a lo largo de toda una vida es insignificante. Créame, el gobierno se lo puede permitir. Recuerde también que toda esta información puede ser analizada en unos segundos para buscar una palabra específica, como por ejemplo marihuana.

Los defensores de la privacidad instan a la población a que evite incorporarse a esa colección de datos mundial de las comunicaciones y movimientos en la red. Imagínese que toda su actividad en la red está siendo controlada por un ordenador, cosa que podría suceder perfectamente.

Imagínese que le gusta vestir ropa de goma o algo similar. No me refiero a que la vista, sino a que tiene ese secretillo, al igual que todos tenemos alguno. Ahora imagínese que el gobierno descubre su secreto porque están escaneando constantemente sus actividades en la red. Saben todo lo que sus ojos miran, cuánto tiempo observan cada cosa mientras está conectado. Con sólo un vistazo cada año a fotos de trajes de buzo, mandiles de carnicero, objetos de látex y la ocasional visita al sitio de los neumáticos Goodyear. Pequeños vistazos a este material 1.480 veces al año lo califican de no muy espabilado y de fetichista.

Con el paso del tiempo, construirán una base de datos que les cuente todo sobre sus inclinaciones. ¿Quiere que suceda esto? ¿Le gustaría que el gobierno, por el significativo precio de 95 pesetas una vida entera, archivara una descripción completa y exacta de su personalidad?

9.2. Contraatacamos

Es su teclado. ¿Por qué son capaces de observar lo que está tecleando? Es su conexión privada a Internet. ¿Por qué

iba a permitirles que investigaran qué sitios ha visitado y qué correo electrónico ha enviado?

Puede tomar medidas para protegerse del "ciberespionaje". En el resto del capítulo le decimos cómo.

9.2.1. Privacidad P3P

En el año 2001 es posible que esté disponible un navegador "plug-in" llamado *Platform for Privacy Preferences* (Plataforma por las preferencias de privacidad, P3P) que debería resolver el problema de "las cláusulas de privacidad". Desarrollado por la organización sin ánimo de lucro *World Wide Web Consortium*, esta ingeniosa utilidad le permitirá especificar el grado de privacidad que quiere mantener mientras navega. Si las cláusulas de un sitio Web están por debajo de su umbral de tolerancia, se le notificará antes de que se descargue en su ordenador. Puede especificar cosas como: "Rechazo visitar sitios que vendan información sobre mí, que proporcionen a otros mi dirección de correo electrónico o que almacenen datos sobre las páginas que visito (lo que miro y por cuánto tiempo)". Para más información acerca de este proyecto puede visitar la página www.w3.org/P3P.

9.2.2. Cuentas de correo electrónico desechables

Cuando contrata un proveedor de Internet (ISP), este le proporciona una o más cuentas de correo electrónico. Esta pasa a ser la más fácil de usar y la identidad primaria que usa cuando se comunica a través de Internet. Sin embargo, también puede contratar cuentas de correo gratuitas (elija Hotmail de Microsoft, Yahoo Mail, Broadcast America o HushMail que le ofrece servicios de criptografía). Al ser gratuita, puede asumir una identidad creando una cuenta de correo electrónico especializada que, por ejemplo, sólo utilice cuando discute en un grupo de noticias políticas, expresando sus opiniones acerca de Cuba o lo que sea. Si, mejor dicho, cuando su visión enfervorizada provoque un masivo ataque de correos violentos, de correos basura o de otro tipo a esa cuenta, no tiene más que cancelarla y contratar una nueva. Tenga en cuenta, sin embargo, que algunas personas encuentran maneras de saber su verdadera identidad aún cuando esté usando una cuenta de correo alternativa.

9.2.3. Reenvíos anónimos (*Remailers*)

Estos servicios son el equivalente en Internet a hacer una llamada telefónica desde una cabina pública; si alguien localiza la llamada, se encontrarán con la cabina vacía. No está allí. Utilice uno de esos servicios de reenvío y su verdadera identidad quedará totalmente oculta. Algunos reenvíos son gratuitos, mientras que otros cobran una cuota o se promocionan.

Los reenvíos anónimos son sitios Web que usan utilidades para sacar su dirección y su nombre de su correo electrónico y utilizan datos falsos de forma que nadie pueda averiguar quién envió el mensaje. Esta técnica, conocida también como *cloaking* o encubrimiento, la usan mucho las personas que envían correo basura, pero nosotros, los legales, también podemos usarla. Quizá tenga posturas políticas radicales, o simplemente quiere expresar su opinión en grupos de noticias sin que se revele su identidad. Al utilizar este servicio, lo que hace es ocultarse.

Si alguien responde a su dirección y nombres falsos, también se le da un nombre falso, pero se le envía a su dirección real que queda memorizada en una base de datos de sitio Web del servicio de reenvío. Esta base de datos vincula su ID falso con los verdaderos datos.

Estos servicios se quedan con el correo electrónico durante una cantidad de tiempo aleatoria antes de enviarlo. Esta característica ayuda a ocultar la fuente porque no hay ninguna relación temporal causa-efecto entre la actividad de su ordenador y el servidor del servicio de reenvío.

Existen dos tipos: los que son realmente anónimos y los pseudoanónimos. Este último es el que guarda en una base de datos sus datos falsos con los verdaderos porque las personas que gestionan el servicio pueden saber su identidad y ser forzadas por los jueces u otros a revelarla (ya ha sucedido). El verdaderamente anónimo no conoce su identidad pero este tipo de servicio no es tan recomendable.

Busque un servicio de reenvío que sea fácil de usar, que lleve en el negocio algún tiempo, que le permita obtener su correo sin que pase a través del proveedor de Internet y utilice una buena criptografía.

Epic (Electronic Privacy Information Centre) es una fuente excelente y fiable para diversos tipos de protección. Búsquelos en www.epic.org/privacy/tools.html.

Junto con esta lista de buenos *remailers* anónimos, encontrará también herramientas que le ayudarán a: navegar por

Internet sin que nadie detecte sus movimientos, sacar todos esos anuncios parpadeantes, librarse de cookies, usar su ordenador como un teléfono seguro, cifrar mensajes y archivos en su disco duro y eliminar archivos de disco y otras utilidades e información.

Si quiere ver otras tácticas para mantener la privacidad de su correo electrónico, visite los siguientes sitios Web:

1. www.obscura.com.
2. www.ziplip.com.
3. www.privatemessenger.com (servicio de criptografía de correo electrónico que incluye cuentas numeradas, gestión de clave automática, based offshore, de forma que no puede ser obligado a revelar información en un juzgado).
4. www.safemessage.com (criptografía, estatus de envío y mensajes que se autoeliminan).
5. www.hushmail.com.
6. www.gilc.org/speech/anonymous/remailer.html.

9.3. La mayor seguridad

Para tener el mayor grado de seguridad posible (¿Es muy, muy paranoico?) puede usar Mixmaster. Esta utilidad construye una "concha" para proteger el mensaje con multitud de criptografías, y a continuación, lo transmite a través de múltiples *remailers* sacando de cada vez una capa diferente. Esto es lo máximo que podrá obtener en seguridad. Busque Mixmaster en la dirección: <ftp://mixmaster.anonymizer.com>.

O consulte estos grupos de noticias para buscar más información sobre Mixmaster y las tecnologías relacionadas:

1. alt.privacy.
2. alt.privacy.anon-server.
3. alt.anonymous.

9.4. Navegar en privado

Cuando visita un sitio Web, las partes interesadas (administración, empresas o individuos) pueden decir en el momento muchas cosas sobre la persona que navega: dónde está, su nombre, tipo de PC, dirección de correo electrónico, qué

navegador usa, qué páginas visita, qué fotos mira y por cuánto tiempo. También las cookies de su ordenador contienen información considerable sobre sus visitas pasadas y otros muchos datos. ¿Hay alguna forma de navegar sin dar toda esa información acerca de nuestra persona? Desde luego.

9.4.1. Navegación privada con *Anonymizer*

Vaya a www.anonymizer.com y escriba una dirección de página Web en el cuadro de texto de **Surf Anonymously**. A continuación, haga clic en el botón de la página principal de *Anonymizer.com*. Verá una segunda página en la que puede introducir bien un nombre de usuario y una contraseña (si es un cliente que paga por los servicios) o hacer clic sobre el botón de **Surf for Free** (navegar gratis). El servicio gratuito tarda en cargarse; también se puede subscribir al servicio premium por 3.000 pesetas por trimestre y evitar de esta forma ese tiempo de carga.

Llegados a este punto, no se podrán saber los sitios que visita o las fotos que examina de cerca ni tampoco incluirán en una lista de "características y preferencias" que la gente construye en sus desagradables bases de datos.

Puede viajar así invisible a través de Internet. En la parte superior de cada página Web verá el cuadro de *Anonymizer Go* disponible por si quiere seguir navegando de forma confidencial. *Anonymizer* ofrece varios tipos de servicios de seguridad, alguno gratuitos, otros no. Haga clic sobre el botón **Services** de su página principal para obtener más información sobre criptografía y otras características.

9.4.2. Confidencialidad con libertad

* Vaya a www.zeroknowledge.com y pruebe *Freedom*, un servicio de privacidad descrito por sus autores como "... una solución en Internet que proporciona una privacidad individual completa cuando se navega, se envía correo electrónico, se chatea o se participa en grupos de noticias dentro de la Web". También describen sus servicios diciendo "Freedom combina pseudónimos durante la navegación, una potente criptografía y tecnología de red para proporcionarle lo mejor en seguridad personal en Internet".

Merece realmente la pena probar Freedom. Es un programa que cuesta 9.000 pesetas y posee muchas utilidades y, al

contrario que otros servicios de privacidad, Freedom ha sido creada para ser doblemente segura: no puede ser citada para revelar su información privada ni ningún empleado chismoso de Zero-Knowledge puede husmear en sus datos. ¿Por qué no? Porque han diseñado el sistema para evitar cualquier asociación, por ejemplo, entre el número de su tarjeta Visa y un seudónimo temporal que pueda querer usar.

La utilidad Freedom se ejecuta de forma transparente al usuario mientras está en Internet, encubriendo las direcciones IP de los comunicadores fuente y de destino y cifrando el continuo flujo de datos entre su ordenador y otros. Zero-Knowledge ha elegido usar buenos sistemas de codificación, como DSA con claves de 1.024 bits y Blowfish, de 128 bits. Créame. Son sistemas potentes. Merece la pena invertir las 9.000 pesetas.

***Nota:** Los sistemas de criptografía actuales refuerzan su seguridad añadiendo bits a las claves. Una clave larga dificulta su desciframiento por parte de un intruso. Cada bit que añade a la longitud de su clave duplica el número de posibilidades. Un sistema de criptografía muy conocido y gratuito es Pretty Good Privacy, que tiene la habilidad de construir claves de 2.048 bits. Sin embargo, como veremos más adelante, es inútil aumentar el tamaño de la clave si se construye un ordenador cuántico.*

Aquí tiene algunos sitios más donde ofrecen técnicas para navegar de forma anónima:

1. www.the-cloak.com (proxy de anonimato, criptografía, guardar cookies para el sitio correspondiente y eliminarlas al acabar la sesión).
2. www.aixis.net (navegación anónima).
3. www.rewebber.de (diversas variedades de protección de la privacidad).

***Nota:** La mayoría de la gente cree que, cuando están chateando, lo que dicen es privado y confidencial, como si fuera una conversación de teléfono ordinaria. Nada que ver. Apenas se han aprobado leyes que regulen el ciberespacio. Algunos servicios de Internet graban y guardan toda la comunicación que se genera en su sistema. Además, algu-*

nos servicios proporcionan utilidades que permiten a los usuarios grabar los chats. Es más, cualquiera puede hacer capturas de pantalla.

ET llama a casa

Se les llama programas ET porque llaman a casa. Aquí tiene la historia de un par de ellos bien conocidos.

RealNetworks ofrece un conocido grupo de programas multimedia, entre los que figuran RealPlayer, RealJukebox, RealDownload y otras utilidades y aplicaciones. Algunas versiones del software son gratuitas, otras no, pero la mayoría de la gente cree que los productos de RealNetworks son útiles y están bien diseñados.

De hecho, empresas como el New York Times, rechazan ofrecer archivos de audio y de video en su página Web que sean compatibles con Windows Multimedia Player. Si quiere escuchar o ver estos archivos debe instalar estos accesorios de Real.

Bajo el manto de las aplicaciones Real, existen, sin embargo, algunas características que han suscitado dudas entre la gente, características que ciñen la idea de personalización más allá de lo que la gente podría desear.

Por ejemplo, si indica sus emisoras de radio favoritas en RealPlayer, estas emisoras aparecen automáticamente en RealJukebox. Si le indica a una de sus aplicaciones sus preferencias sobre música, películas, deportes y software la "guía de entretenimiento diario" le envía noticias sobre productos o desarrollos que se están llevando a cabo en las áreas que ha marcado como preferentes.

Hasta este punto quizás sea aceptable. Pero algún software previo de Real incluía un elemento denominado GUID (Global Unique Identifier, Identificador único mundial) del que no se informaba a los usuarios. Más adelante, algunos usuarios se quejaron de que muchas URL (direcciones de Internet) eran reenviadas a RealNetworks. Las preferencias del usuario pueden agruparse de esta manera y la publicidad puede adecuarse a ellas. Real dice que la información personal no está aho-

ciada con la información URL y que esta última no se retiene; todo sucede en tiempo real. Este enfoque impide claramente que se cometan abusos porque no se archivan los datos.

La última versión del software de Real también incluye un GUID pero, por defecto, está desactivado. Los usuarios pueden elegir la opción de activarlo si quieren ver, por ejemplo, programación de pago.

Aunque las características de personalización de Real sean inocentes y útiles, hay muchas empresas por ahí, incluidos los ISP, recogiendo este tipo de datos:

Por ejemplo, zBubbles, una práctica utilidad del software para navegadores Alexa (propiedad de Amazon), ha resultado ser útil en más de una forma. El usuario se beneficia porque, si encuentra algo de oferta en una página Web, zBubbles abre un cuadro de diálogo y le explica en qué consiste ese elemento y también si lo puede encontrar más barato en otra parte; a lo mejor, es más barato si lo compra en Amazon.

Se ha informado de que zBubbles no se limita en ayudarle a conseguir las mejores ofertas en la red. Algunas personas afirman que si, por ejemplo, está pensando comprarse unos CD, los títulos específicos de estos junto con su dirección de correo electrónico le son enviados a zBubbles a su sitio Web. ET llama a casa. Al igual que RealNetworks, los propietarios de zBubbles dicen que la información que se recoge no está correlacionada con usuarios particulares.

El software ET es una marea que sube y puede que muchas compañías no sean tan escrupulosas con sus políticas de privacidad como RealNetworks y Amazon. No es el único al que le desagrada la idea de que le pongan un pequeño programa en su disco duro, cuyo único propósito es enviar información a su propietario sobre sus datos de navegación y compra. Todo esto puede suceder sin su consentimiento y sin que ni tan siquiera lo sepa. Por el momento, este tipo de cosas sigue siendo legal. ZoneAlarm, BlackICE u otros cortafuegos similares pueden instalarse para informar acerca de cualquier intento de enviar información desde su ordenador a la red. En el caso de ZoneAlarm, si un programa intenta acceder a Internet, aparece un cuadro de diálogo que le informa de qué programa está intentando enviar datos y se le pide

permiso antes de hacerlo. De esta forma, ZoneAlarm le indica cuándo se está desarrollando una actividad ET en su máquina y también le permite bloquear esas llamadas.

Si su módem tiene luces, se dará cuenta de que, de vez en cuando, parpadea una de ellas. Puede estar indicando que ET llama a casa. Se está enviando información desde su disco duro a un sitio Web. A veces, sucede esto cuando ni tan siquiera está usando el ordenador, sin teclear, sin tocar el ratón. Pero está ahí: la luz parpadeante.

Reflexione sobre las consecuencias que esto tendrá cuando se usen las comunicaciones sin cable. No sólo estará activada siempre su conexión a Internet sino todo su ordenador estará constantemente activado. Con los sistemas de localización por satélite, es posible que alguien pueda saber en qué punto de la tierra está en todo momento. Estas y otras posibilidades son el motivo por el cual los defensores de la privacidad reclaman una legislación que limite duramente las transmisiones de ET, la generación de perfiles personales y otra protección de la privacidad.

Aparentemente, ni siquiera los productos infantiles son inmunes a estos pequeños trucos de retroalimentación. USA Today informó recientemente de que algunas personas se quejaban de que algunos de los códigos que venían incluidos en el software infantil de Mattel (por ejemplo, la serie de Reader Rabbit) estaba diseñado para enviar información a Mattel. Mattel Interactive dijo que propocionaría una utilidad denominada "Broadcast". Esté al loro.

9.5. También analizan su teclado

Aún cuando cifre y reenvíe su correo electrónico, utilice un cortafuegos potente y efectivo, navegue a través de redireccionadores y tome otras precauciones prudentes para proteger su privacidad, puede seguir siendo víctima del espionaje. Los controladores del teclado graban cada pulsación que da, así que obtienen información de la fuente fidedigna. Cuando escribe, la información está en la más pura de sus formas. Todavía no ha sido cifrada, reenviada ni escondida de ninguna otra forma. Algunos de estos

controladores también almacenan imágenes de la pantalla en intervalos regulares.

Disponibles desde hace quince años, los programas de control de teclado solían almacenar toda la información de las pulsaciones en un archivo en el disco duro. Pero en la actualidad, se envían estos datos de vuelta a través de Internet (estilo ET), bien a través de correo electrónico o bien se cargan directamente. Cualquier nombre de usuario o contraseña que escriba, junto con los URL, números de tarjeta de crédito y cualquier otra cosa puede guardarse y enviarse a un hacker, whacker o a su jefe. Es legal, ya lo sabe, que su empresa mire qué es lo que teclea en *sus* máquinas.

9.5.1. Defendámonos

Puede instalar cortafuegos como ZoneAlarm para que se le indiquen las llamadas salientes. Pero los cortafuegos le dan la posibilidad de activar la concesión de salida de llamadas a ciertos programas. De esta forma, no tiene que dar su permiso cada vez que use su correo electrónico o el navegador, por ejemplo. Sin embargo, si el controlador del teclado usa su registro para incorporarse como software del navegador o del correo electrónico, sus pulsaciones pueden ser enviadas de forma invisible porque ya ha dado su permiso. Si sospecha que esto sucede, intente activar ZoneAlarm para que tenga el máximo control de seguridad con la información saliente (no se permite nada), y vea qué tipo de tráfico choca contra la barrera de ZoneAlarm y este le advierte de lo que pasa.

9.5.1.1. Chequeo de Inicio

Mucha gente es consciente de que pueden instalar aplicaciones y utilidades en la carpeta de Windows de Inicio, de forma que se ejecuten cada vez que se inicie Windows. (Ponga el puntero sobre **Inicio**, haga clic con el botón derecho. Elija **Explorar**. Localice la dirección **Windows, Menú Inicio, Programas**, carpeta **Inicio**.)

Sin embargo, poca gente sabe que un gran grupo de utilidades se cargan cuando se inicia Windows, pero no aparecen listadas en la carpeta Inicio. Entre ellas puede estar el controlador del teclado. Puede desactivar selectivamente cualquiera de estas utilidades escondidas de inicio para ver

el efecto. También puede echar un vistazo a los nombres de los programas y su localización en el disco duro. Si encuentra uno que se llama *Keystroke checker*, *winwhatwhere*, o algo parecido, puede querer desactivarlo. Para saber exactamente qué es lo que se está cargando en su ordenador cuando lo enciende, siga los siguientes pasos:

1. Haga clic sobre el botón **Inicio**.
2. Escoja la opción **Ejecutar** del menú **Inicio**.
3. Teclee **msconfig** y acepte.
4. Haga clic sobre la pestaña **Inicio** en **Programa de Configuración del Sistema**.

Verá algo similar a la figura 9.1

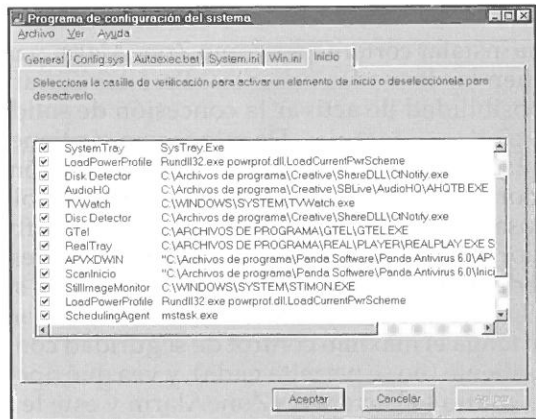


Figura 9.1. Aquí es donde puede especificar exactamente los programas que se ejecutan cuando inicia Windows, y quizá descubrir un controlador de teclado pululando entre los verdaderos programas útiles como System Tray

9.5.1.2. Buscar archivos para una palabra concreta

Otra forma de encontrar controladores de teclado es enviarse un correo a uno mismo y ver si el mensaje queda almacenado en un archivo extraño. Siga estos pasos:

1. Ejecute su programa de correo electrónico y escriba correo nuevo.

2. Envíe el mensaje a su propia dirección.
3. En el cuerpo del mensaje escriba la extraña palabra *tessor*.
4. Ahora rastreará todo su disco duro para ver si encuentra esta palabra, así que vaya a **Inicio, Buscar, Archivos o Carpetas**.
5. Escriba en el campo **Con el texto** la palabra *tessor*.
6. En el campo **Buscar en** elija C: o la etiqueta correspondiente a su disco duro.
7. Haga clic sobre **Buscar Ahora** (buscar en cada byte de cada archivo de su disco duro le puede llevar algo de tiempo).

La palabra aparecerá en el archivo de Bandeja de Entrada de su programa de correo, así como en Elementos Enviados; lo ha enviado y lo ha recibido, así que la palabra *tessor* debería aparecer en ambos archivos. Si aparece en otros, sospeche y chequee esos archivos (léalos en el Bloc de Notas, WordPad o cualquier procesador de texto). Si ha encontrado la palabra en un sitio inesperado, probablemente habrá localizado un registro de un controlador de teclado.

9.6. La criptografía es una potente defensa

Si alguien graba todas las pulsaciones que da sobre el teclado, no es buena idea cifrar sus mensajes. Los datos ya han sido grabados antes de que tenga oportunidad de ocultarlos.

Pero si nadie graba información de su teclado, una de las mejores defensas que puede levantar contra la invasión de su privacidad es usar el antiguo arte de la criptografía. Oculte su información. De ese modo, aunque entren en su disco duro y cojan sus archivos, no podrán leerlos.

Los siguientes capítulos del libro exploran la ciencia cambiante de la criptología, que consiste en transformar información (cifrándola) y, por lo tanto, que sea inútil a cualquiera que no sepa descifrarla.

La criptografía juega un importante papel en el comercio electrónico. A mucha gente le preocupa dar sus datos personales, especialmente números de Visa y similares, a través de Internet. Si empieza a pagar facturas en la red, quiere estar seguro de que sólo los empleados del banco correspon-

dientes, y usted, pueden entrar en sus cuentas para manipular el dinero. No quiere que haya extraños metiendo las narices en su cuenta y que suelten esta información libremente por toda la red.

Windows 2000 y Mac's OS9 tienen buenos sistemas criptográficos incorporados al propio sistema operativo. Son rápidos, adecuados y fáciles de usar. Pero si usa un sistema operativo diferente, o prefiere otros sistemas criptográficos más potentes, encontrará todo lo que necesita en los siguientes capítulos.

Demasiadas pistas y huellas

¿Qué es lo que hace cuando sustituye el viejo ordenador por uno nuevo? ¿Se lo da a un amigo o lo deja a la buena de Dios? ¿Lo vende o intercambia? ¿Tiene la precaución de borrar archivos y carpetas? ¿Se le ha ocurrido formatear el disco duro?

Poca gente sabe que borrar archivos no significa eliminarlos y dejar el disco limpio como cuando se borra una pizarra. Lo que sucede es similar a destruir los archivos de catalogación de una biblioteca: aunque lo destruya, los libros se quedan ahí en sus estanterías. Cuando borra archivos, sólo borra el índice de esos archivos; los datos en sí se quedan en el disco.

Su disco duro contiene mucha información personal. Sus movimientos en Internet, páginas que ha visitado, cookies que ha recogido, correo electrónico que ha enviado y recibido; todo está archivado en el disco duro. Además, seguro que tiene almacenada otra información confidencial, como sus cuentas, números de tarjeta de crédito, contraseñas, declaraciones de la renta, cartas personales, etc.

Cuando cambie su ordenador, hay varios pasos que debe seguir para limpiar el disco duro viejo. Guarde su libro de direcciones de correo electrónico y sus favoritos del navegador en un disquete. Utilice las opciones de Internet Explorer Archivo, Importar y Exportar,. A continuación elija Archivo, Exportar, Libro de direcciones en Outlook Express. A continuación, desinstale su programa de gestión de correo electrónico y el navegador. Haga una co-

pia de seguridad de todos los archivos .DOC y otros archivos de datos que quiera llevar al nuevo ordenador. A continuación, puede usar uno de los muchos programas comerciales para dejar el disco duro completamente limpio. Pruebe con CyberScrub (está en el CD del libro). También está el programa llamado WipeInfo, de las Utilidades Norton, Infracore's Sanitizer o OnTrack's DataErase. Estos programas pueden sobrescribir cada byte de su disco duro.

X

Los elementos de la criptografía

"Mira", dijo el hombre de Neandertal a su compañero de caza, "no voy a gruñir cuando vea al gran mamut, porque escapará corriendo como la última vez. Así que, cuando ponga la mano en la cabeza, significará 'arroja la lanza', ¿lo has entendido?"

Este primer código fue algo así como un acuerdo entre los primeros cazadores, pero la gente aún utiliza versiones de esto en fiestas: "Cuando empiece a hablar de las vacaciones del año pasado, significa que nos vamos, ¿vale?"

Códigos, criptografía, mensajes cifrados, esteganografía, son muchos los nombres para referirse a un mensaje de una a otra persona sin permitir que los demás lo entiendan (el mamut, la anfitriona, etc.)

10.1. Codificar *versus* Cifrar

Empezaremos por algunas definiciones. Los expertos señalan varias diferencias entre las diversas técnicas usadas para enviar mensajes secretos. La *esteganografía* trata la ocultación física de un mensaje (micropuntos; esconder un trozo de papiro en la barriga de un conejo muerto, hacer que un cazador lo lleve hasta el siguiente castillo con el animal colgado a la espalda; reventar transmisiones de radio; tinta invisible y otros métodos que no permiten ver que existe un mensaje).

La criptografía no esconde el hecho de que se está enviando un mensaje, pero intenta hacerlo incomprensible transformando el original. (El original, el mensaje que se puede leer, se denomina texto plano y el texto cifrado.)

La criptografía en sí se puede dividir en dos técnicas principales: el cifrado y la codificación.

El cifrado consiste en cambiar las posiciones de las letras de cada palabra (transposición) o sustituye las letras con símbolos (sustitución). Los cifrados manipulan letras del abecedario o grupos de letras.

La codificación, sin embargo, trabaja más con caracteres que con letras. Un código sustituye grupos de palabras al mismo tiempo. Un código está formado por una gran lista de palabras con su correspondiente lista de palabras código (o símbolos). Por ejemplo podría crear su propio código cogiendo un diccionario, asignarle una palabra clave a cada palabra y escribirlas juntas. Por ejemplo:

<i>Texto plano</i>	<i>Su código</i>
boa	Marat
bobera	xea#
bobillo	411
bobina	axptor!

La distinción más práctica entre códigos y cifrados es que estos últimos funcionan sólo manipulando letras individuales y los códigos manipulan las unidades mayores del lenguaje, como sílabas, palabras o, incluso, frases enteras.

Hemos estado hablando hasta ahora de enviar mensajes secretos y, por supuesto, eso es tan útil hoy en día en Internet como lo fue en las antiguas civilizaciones en las grandes guerras y en el comercio. Pero existe otro uso de la criptografía que no implica el envío de secretos entre dos personas. ¿Ha escrito alguna vez un diario con información personal que no quería que nadie viese? La criptografía es una tecnología fundamental y duradera con muy diversas aplicaciones. Por ejemplo, cifrar el número de su tarjeta Visa antes de enviarlo a través de Internet o cifrar sus datos económicos personales residentes en el disco duro.

Tecnología es probablemente una palabra errónea para describir las técnicas antiguas. Cuando los emperadores chinos escondían sus mensajes secretos en pequeñas bolas de cera escondidas en las partes privadas del cuerpo de los mensajeros, era un tipo primitivo de esteganografía. Sin embargo, en las últimas décadas, la criptografía ha avanza-

do considerablemente y hace uso de técnicas que sólo los ordenadores pueden usar. Cuando se aplicó la tecnología a este antiguo arte, hubo un gran progreso, tanto en las técnicas de cifrado como en las de descifrado.

10.2. Una antigua perfección

Aunque la criptografía ha visto ya miles de años de mejora y un salto considerable gracias a los ordenadores, hay una antigua técnica que no puede ser descifrada ni por las mejores técnicas actuales. No importa lo potente que sea el ordenador, la técnica denominada *one-time pad* (bloc de notas de un único uso) desafía el criptoanálisis y el desciframiento.

En 1994, Evangelos Petrosus y yo escribimos un libro titulado *The Visual Basic Power Toolkit*, que incluía un reto. Describimos detalladamente una técnica criptográfica, proporcionamos un pequeño programa de ordenador que usaba la técnica e incluimos un mensaje secreto que generamos usando la técnica.

Ofrecimos 200.000 Ptas. a la primera persona que fuese capaz de resolver el mensaje, pero jamás lo consiguió nadie. Como veremos más adelante, existe una forma perfecta de cifrar mensajes. Sin embargo, en la Segunda Guerra Mundial, las potencias aliadas y del Eje enviaban más de 60 millones de palabras cifradas al mes. La técnica perfecta criptográfica que se describe más adelante no es práctica para semejante escala de tráfico. Es decir, es perfecta para criptografía a baja escala, para su información personal secreta; a este nivel, su comportamiento es intachable.

No me importaría que consiguieran diseñar la máquina definitiva de criptoanálisis.

10.3. Cómo descifrar mensajes secretos

Desde que unos empezaron a esconder información otros, por supuesto, intentaron descifrar los códigos y mensajes usados.

Intente resolver este mensaje cifrado basado en lengua inglesa.

Ab rtx cxor adrxsxoro mp rtx qxmqux

Si tiene virtudes para el criptoanálisis, se dará cuenta de que una de las palabras está repetida: *rtx*. Y, si ha leído algo sobre esta ciencia, sabrá que la palabra más habitual en inglés es el artículo *the* y que la letra más habitual es la *e*, concretamente, una de cada ocho.

La *x* es el carácter más repetido en el texto cifrado por lo que podemos intuir que se corresponde a la *e*. Además, esto también encaja con la posibilidad de que *rtx* sea *the*. Si sustituimos *e* por *x* y *the* por *rtx*, obtenemos:

Ab the ceor abreseoro mp the qemque

Ahora, si asumimos que *rtx* es *the*, podemos sustituir *t* por todas las *r* que aparecen en el texto cifrado:

Ab the ceot abteseoto mp the qemque

Si continuamos nuestras pesquisas, podemos usar los grados de frecuencia de las letras y palabras inglesas. Las nueve letras más frecuentes en cualquier intervención en inglés son: *e, t, o, a, n, i, r, s* y *h*.

Un veinticinco por ciento de las palabras usadas en inglés son: *the, of, and, to, a, in, that, it, is* y *I*.

Basándose en esta información, puede descifrar el mensaje rápidamente:

In the best interests of the people (Por el bien de la gente.).

A continuación, tiene la tabla de frecuencias, con porcentajes, de las seis letras, grupos de letras y palabras que más se usan en inglés americano.

Letra	Grupos de 2 letras	Grupos de 3 letras	Palabras completas
E (13%)	TH (3%)	THE (5%)	THE (6,5%)
T (9%)	IN (1,5%)	ING (1,5%)	OF (4%)
O (8%)	ER (1,3%)	AND (1%)	AND (3,1%)
A (7,8%)	RE (1,3%)	ION (1%)	TO (2,3%)
N (7,2%)	AN (1%)	ENT (0,98%)	A (2%)
I (7%)	HE (1%)	FOR (0,76%)	IN (1,77%)

Tabla 10.1. Frecuencias en inglés norteamericano

10.4. La gente utiliza trucos

Obviamente, una simple sustitución de caracteres como la que acabamos de analizar no funcionaría mucho en la actualidad. Mucha gente sabe demasiado sobre descifrar. Consecuentemente, durante décadas, la gente ha ido complicando más y más las formas de oscurecer el significado de sus mensajes secretos. Además de las primeras herramientas de cifrado (sustitución y transposición), hay otros tres trucos secundarios: expansión, compresión y división en bloques.

Una buena idea es cubrir el texto cifrado con *nulls*, que son caracteres que no tienen significado pero que se insertan en el mensaje. El emisor y el receptor del mensaje conocen la lista de caracteres falsos, así que los ignoran. Si insertamos aleatoriamente *nulls* como estos (*#@!z*), el mensaje visto en el otro ejemplo sería mucho más difícil de descifrar y tendría el siguiente aspecto:

Ab !! rt@x cx#or! Abzrx#sxo | r@o mzp rtzx q@!xmqux#

Sin embargo, incluso este caos es relativamente fácil de resolver usando un análisis de frecuencia. La adición de *nulls* y otras formas de engaño se denomina expansión. *Pig Latin* es un crudo ejemplo de expansión.

Una técnica relacionada con la expansión es la denominada compresión, contracción o compactación. En este caso, en vez de falsear el texto plano, lo reduce. Un ejemplo simple es sacar los espacios que hay entre las palabras; sigue siendo fácil leer un mensaje después de eliminar los espacios. Puede eliminar la puntuación, sin causar una seria pérdida de información, aunque, a veces, marca una diferencia significativa como en: *Pedro, mi amigo ha llegado y Pedro, mi amigo, ha llegado.*

En otras variedades de compresión, se extraen partes del mensaje original y se transmiten más tarde, separados del mensaje principal. Se acuerdan previamente las normas que describen qué partes se transmiten primero, de forma que el receptor puede reconstruir el texto original de acuerdo con ellas.

Una técnica final, muy popular actualmente, es la división en bloques. Toma el mensaje original, divide los caracteres en grupos de, pongamos, ocho caracteres. A continuación, manipula esos bloques independientemente. Esto le permite realizar sustituciones, permutaciones o transformaciones matemáticas en cada bloque en vez de en el conjunto

del texto original. Una ventaja significativa de esta técnica es que puede aplicar los resultados de la transformación de un bloque al siguiente bloque.

Por ejemplo, tras haber hecho varias sustituciones en un bloque de ocho caracteres, podría sumar los valores de los caracteres de todos los bloques anteriores ya cifrados y, después, añadir el valor al tercer carácter del bloque con el que va a trabajar. A los caracteres alfabéticos se les pueden asignar valores numéricos arbitrariamente, así que puede trabajar con estos valores: A=1, B=2, etc. La virtud primaria de conseguir que un aspecto de cada bloque dependa del resultado del bloque anterior, es que el proceso criptográfico completo pasa a ser altamente interdependiente. Si cambia un solo carácter en el mensaje original, podría originar que cambiasen todos los caracteres subsecuentes en el texto cifrado. Veremos la división en bloques empleada en el famoso sistema DES descrito con detalle más adelante.

Afortunadamente, para aquellos de nosotros que encontramos la ciencia del cambio de la información fascinante, la criptología no ha sido una disciplina estática en los últimos 10.000 años. Por ejemplo, veremos más adelante en este libro que hubo una gran ruptura en el siglo XV, cuando alguien tuvo la idea de usar un abecedario diferente para cifrar cada letra del mensaje original. Obviamente, esto distorsiona gravemente las frecuencias de los caracteres.

Un punto importante: cuantos más secretos transmita usando el mismo sistema, más vulnerable se vuelve este. Durante una típica guerra mundial, se transmiten al día varios cientos de miles de palabras cifradas. Es precisamente la transferencia de esta inmensa cantidad de información la que marca los tipos de cifrado que son prácticos para operaciones a larga escala como las guerras. Los modelos de cifrado más robustos son efectivos en operaciones a pequeña escala (como la fascinante, pero pequeña, información de su diario). Más adelante trataremos un sistema tan perfecto que ni su brillante cuñada criptoanalizadora podría descifrar en toda una vida de trabajo.

10.5. El objetivo de la criptografía

¿Cómo conseguir que un modelo irreconocible se pueda recuperar? Si una mínima parte de la información del mensaje cifrado permanece inalterada, un intruso podría usar

potencialmente esa pequeña pieza para reconstruirlo. Usaremos la palabra intruso para referirnos a alguien que se apropia de su mensaje codificado e intenta descifrar el código. El etiqueta *Eve* también se usa en este ámbito para el mismo propósito).

Tiene que hacer algo para distorsionar y mezclar el mensaje. Sin embargo, cuando distorsiona algo, (por ejemplo, cuando aplasta las patatas con el tenedor), crea un nuevo modelo: la forma del instrumento distorsionante. En este ejemplo sería el tamaño y la posición de los dientes del tenedor. Un enemigo que sepa reconstruir la herramienta que usó para distorsionar el mensaje puede deducir el modo de realizar el proceso inverso a través del caos de caracteres cifrados y, consecuentemente, reconstruir el mensaje original.

El primer trabajo de un descifrador intruso es buscar repeticiones. La repetición revela modelos, texturas. El intruso puede comenzar por investigar modelos típicos del idioma, como las estadísticas de frecuencias de letras y palabras.

Puede pensar: "¿Qué tal si confundo al intruso alterando realmente las cosas? No sólo haré sustituciones e introduciré nulls, sino que también utilizaré normas de transposición raras. Por ejemplo, cambiaré el orden cada cuatro letras de forma que quede el modelo: 1 2 4 3 5 6 8 7...".

Puede mejorar esquemas simples incluyendo este tipo de complicaciones, pero los ordenadores pueden chequear rápidamente millones de posibles sustituciones y transposiciones. Pueden analizar su mensaje hasta que aparece la palabra *the* o alguna otra pista sobre el texto original. Y sólo necesitan unos segundos para hacerlo. Recuerde que pocos seres humanos son capaces de vencer ya a un ordenador jugando al ajedrez.

Mi ordenador, un Pentium III a 550 MHz puede contar de 0 a 11.696.443 en un segundo usando el siguiente programa:

```
Dim x As Long
Private Sub Form_Load()
    rep:
    x = x + 1
    GoTo rep
End Sub
```

Incluso antes de que existiesen ordenadores, hubo gente que, con suficiente tiempo libre e ingenuidad, fue capaz de descifrar rápidamente los modelos básicos criptográficos.

Después hubo un gran salto durante el siglo XV, que es el tema del siguiente capítulo.

El gran salto hacia adelante

Durante los primeros milenios de la civilización humana no hubo grandes avances en la ciencia de la criptología. Se introdujeron algunas variaciones pero no hubo tendencias importantes que la hicieran evolucionar hasta 1466 cuando León Alberti, creador de la Fontana de Trevi, reveló al mundo en uno de sus escritos la teoría de la *sustitución polialfabética*.

11.1. La propuesta de Alberti

Inspirado por la reciente creación de la imprenta, Alberti tuvo la genial idea que sería la base de toda la criptografía moderna. Las técnicas de la sustitución polialfabética utilizan una tabla como esta:

a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z
b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	a
c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	a	b
d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	a	b	c
e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	a	b	c	d
f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	a	b	c	d	e
g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	a	b	c	d	e	f
h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	a	b	c	d	e	f	g
i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	a	b	c	d	e	f	g	h
j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	a	b	c	d	e	f	g	h	i
k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	a	b	c	d	e	f	g	h	i	j
l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	a	b	c	d	e	f	g	h	i	j	k
m	n	o	p	q	r	s	t	u	v	w	x	y	z	a	b	c	d	e	f	g	h	i	j	k	l
n	o	p	q	r	s	t	u	v	w	x	y	z	a	b	c	d	e	f	g	h	i	j	k	l	m
o	p	q	r	s	t	u	v	w	x	y	z	a	b	c	d	e	f	g	h	i	j	k	l	m	n
p	q	r	s	t	u	v	w	x	y	z	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
q	r	s	t	u	v	w	x	y	z	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p
r	s	t	u	v	w	x	y	z	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q
s	t	u	v	w	x	y	z	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r
t	u	v	w	x	y	z	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s
u	v	w	x	y	z	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t
v	w	x	y	z	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u
w	x	y	z	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v
x	y	z	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w
y	z	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x
z	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y

Recuerde que en capítulos anteriores hemos hablamos de todos los métodos previos a éste que usaban la sustitución dentro de un mismo abecedario, por ejemplo, la *x* siempre significaba *e*.

El gran mérito del enfoque polialfabético es que puede usar un alfabeto diferente por cada letra del mensaje. La *x* podría significar algunas veces *e*, pero otras podría representar otro carácter distinto.

Los modelos repetidos (como la palabra alemana *Führer* en la Segunda Guerra Mundial o el artículo determinado *el*, *la*, *los*, *las* en cualquier idioma) se pueden disfrazar con efectividad pero no eliminarse. El análisis de frecuencia, la única técnica potente en el desciframiento de mensajes secretos, se ve así contrarrestado. Pero en criptología, por cada gran sistema que surge, acaba apareciendo un contra-sistema. Al final de este capítulo veremos cómo en 1881 un brillante criptoanalista flamenco diseñó una forma para descifrar la sustitución alfabética.

Esto es un ejemplo que ilustra el funcionamiento del polialfabeto. Eche un vistazo a la tabla. Puede cifrar un mensaje polialfabéticamente usando el abecedario de la parte superior (la primera línea) como referencia del texto plano y luego ir descendiendo una línea por cada letra.

EL ARBOL EL MAR

pasa a ser

EM CUFTR LT VKC

Fíjese que la palabra *el*, que aparece dos veces en el original, está cifrada de dos maneras diferentes, al igual que las tres *l*, las dos *r* y las dos *a*.

Nota: Por supuesto, no tiene ninguna lógica usar espacios cuando estamos cifrando un texto, pues le estaríamos dando al enemigo la ventaja de saber cuántas palabras hay, su extensión y su posición. Pero, por motivos ilustrativos, voy a usarlos en estos ejemplos.

Existen complicaciones adicionales que se han sumado a la técnica del polialfabetismo a lo largo de los años. Una de las más destacadas es usar una clave secreta.

Llamada también contraseña, una clave es una palabra o frase (o mejor aún, una combinación aleatoria de caracteres

y dígitos) que sólo conocen el emisor y el receptor del contenido que se pretende transmitir; es decir, el intruso no conoce la clave. Hablaremos más sobre claves en capítulos posteriores. Por ejemplo, cuanto más larga y aleatoria sea, mejor.

Sin embargo, fíjese hasta qué punto puede complicar el trabajo de un intruso usando una clave junto con el código polialfabético. Digamos que acuerda con su amigo que su clave sea:

BIENPUESTOQUELAOFICI

Puede usar la clave para moverse sobre los abecedarios de la tabla en vez de hacerlo línea a línea, usando el alfabeto siguiente como hicimos en el ejemplo anterior.

Aquí tiene un ejemplo de uso de clave. Localice la primera letra de la clave, *B*, en la línea superior (use esta línea para la clave). A continuación, localice la columna de la izquierda (utilícela para escribir el mensaje original). Encuentre la primera letra de su mensaje (EL ARBOL, EL MAR). Esta letra es *E*. Así, su primer carácter del mensaje cifrado es la intersección donde estos dos caracteres se encuentren en la tabla. Es la letra *F*. A continuación, localice la siguiente letra, *I*, en la fila superior y baje hasta que encuentre la intersección con la *L*. Es la letra *T*. Siguiendo este proceso, el mensaje obtenido sería:

FT EEQIP WE AQL

Es relativamente fácil escribir un programa informático que utilice este sistema para cifrar y descifrar mensajes. Pero es relativamente difícil para las personas (sin ordenadores) descifrar mensajes cifrados con códigos polialfabéticos.

Hasta el momento, hemos usado una tabla que sólo contiene 26 abecedarios (uno para cada letra) que se repiten. Sin embargo, no hay razón alguna para limitar el uso a 26. Ya desde hace algún tiempo, la criptología asistida por ordenador ha estado usando millones de alfabetos. Son realmente *poli*. El uso de tanto alfabetos evita las repeticiones ya que hay más alfabetos que caracteres en el texto original.

11.2. Un difícil experimento

Imaginemos que queremos comunicar una idea que nadie más debe saber. Nos ponemos previamente de acuerdo

en que si digo "No hay muchos concesionarios por aquí", lo que quiero decir es "No soy capaz de conseguir una cita".

Nadie es capaz de traducir esto, ni tan siquiera de saber que se trata de un mensaje secreto.

Puede pensar que este sistema funciona perfectamente. Es cierto, si sólo queremos transmitir una única idea, funciona de maravilla. Pero, ¿cómo hace para decir que ha conocido a una bonita pelirroja o a un morenazo y que ha quedado con ella o con él? No puede hacerlo usando el código de los concesionarios, ¿verdad?

Una vez que usa un lenguaje natural, está introduciendo inevitablemente modelos y repeticiones en el mensaje codificado. Por ejemplo, en períodos de guerra los mensajes deben ser contruidos y destruidos con gran rapidez. Aún así, es predecible que los mensajes contengan palabras del estilo: submarino, misil, etc., que además hay que repetir. Recuerde que la repetición es el punto flaco crucial de cualquier mensaje cifrado. Aunque utilice millones de alfabetos, los ordenadores pueden chequearlos todos sin derramar ni una gota de sudor.

11.3. La segunda gran idea de Alberti

Alberti no se detuvo con su gran contribución de la técnica del polialfabeto. Su segunda idea importante fue combinar los dos tipos primarios de mensaje secreto: cifrar y codificar.

Recordemos que un código es una representación uno a uno de unidades lingüísticas (frases o palabras). Podemos crear la siguiente lista de 2.000 cláusulas y segmentos de cláusulas que tendríamos que usar en el desarrollo de una guerra, asignándoles un grupo aleatorio de letras:

<i>Texto plano</i>	<i>Código</i>
Necesitamos agua	aoqf
Llame a los refuerzos	zvow
Ahora	qwvs
Demasiado calor	vpwz
Van de camino	utyo
Mañana radio	urto

Cada uno de nosotros tendría esta lista (código), y sabríamos lo que significan las distintas agrupaciones de caracteres. El siguiente paso en el plan de Alberti sería aplicar a la frase resultante la técnica del polialfabeto. Supongamos que queremos enviar el mensaje: *Demasiado calor, necesitamos agua*. El código quedaría así.

vpwzaoqf

A continuación, utilizaríamos la tabla del polialfabeto:

vtzbbopd

11.3.1. Un resultado inútil

Aún asumiendo que un intruso fuese capaz de descifrar el cifrado polialfabético, obtendría el siguiente mensaje: vpwzaoqf. Este resultado sería absolutamente inútil a menos que se las ingeniara para conseguir la lista de códigos. Hay que admitir que Alberti también tuvo una gran idea aquí.

Si está interesado en hacer pruebas con los descubrimientos de Alberti, puede usar este pequeño programa de Visual Basic que transforma un mensaje original en un mensaje cifrado polialfabéticamente, usando la tabla que incluimos al principio del capítulo.

```
Private Sub Command1_Click()  
    alpha = "abcdefghijklmnopqrstuvwxyz"  
    plain = Text1  
    If plain = "" Then MsgBox "no message to  
encipher": Exit Sub  
    If Len(plain) > 26 Then MsgBox "this program  
works on messages shorter than 27 characters":  
Exit Sub  
  
    For i= 1 To Len(plain)  
        keyposition = Instr(alpha, Mid(plain, i,  
1)) 'get key position of current plaintext  
letter from top row of tableau  
        If keyposition = 0 Then GoTo blanks'  
space character probably, or other invalid  
  
        'create current alphabet based on iteration  
number
```

```

        leftside = Mid(alpha, i, 27- i)

        rightside = Mid(alpha, 1, i -1)

        newalphabet = leftside & rightside

blanks:

    If keyposition = 0 Then
        cipher = cipher & ""
    Else
        cipher = cipher & Mid(newalphabet,
keyposition, 1)
    End If

Next i

Text2 = cipher

End Sub

```

Para realizar este proyecto con Visual Basic, ponga los dos cuadros de texto (denominados Text1 y Text2) en un formulario y añada un botón (denominado Comand1). Para seguir trabajando a un nivel sencillo, este programa no cifra mensajes de más de 26 caracteres y sólo reconoce minúsculas. Si quiere cifrar mensajes más largos, cambie la propiedad de multiline del cuadro de texto a true y añada índices al código fuente que recoloque los indicadores en el alfabeto en vez de usar la variable i, como he hecho yo en el ejemplo. El código fuente, así como la versión compilada .EXE de este programa aparecen en el CD del libro.

11.3.2. El desciframiento invierte el proceso

El amigo que está al otro lado del proceso de comunicación secreta necesita descifrar el mensaje. Para descifrar este tipo de sustitución polialfabética, se localiza el abecedario correcto en la tabla (en estos ejemplos usamos el alfabeto en sentido horizontal descendente, de la misma forma que cuando ciframos mensajes). Sin embargo, durante el desciframiento, debe buscar la posición del texto actual dentro del alfabeto

y usar esta para encontrar el número de posición para la letra del mensaje original (la línea superior de la tabla). Repita este proceso para encontrar la solución, buscando las letras una a una.

Aquí tiene un programa de Visual Basic que descifra o resuelve mensajes cifrados con el programa visto anteriormente.

```
Private Sub Command1_Click()

    alpha = "abcdefghijklmnopqrstuvwxyz"
    encryption = Text1
    If encryption = "" Then MsgBox "no message
to decipher": Exit Sub
    If Len(encryption) > 26 Then MsgBox "this
program works on messages shorter than 27
characters": Exit Sub

    For i= 1 To Len(encryption)
        'create correct alphabet based on iteration
number

        For j = 1 To i
            leftside = Mid(alpha, j, 27- j)
            rightside = Mid(alpha, 1, j -1)

            newalphabet = leftside & rightside
        Next j

        alphabetposition = InStr(newalphabet,
Mid(encryption, i, 1))
        If alphabetposition = 0 Then GoTo blanks'
space character
        probably, or other invalid

        blanks:

        If alphabetposition = 0 Then
            plaintext = plaintext & " "
        Else
            plaintext = plaintext & Mid(alpha,
alphabetposition, 1)
        End If
    Next i
End Sub
```

Next i

Text2 = plaintext

End Sub

Ahora, con estos dos programas, tiene un sistema de cifrado/descifrado. Es lo suficientemente bueno como para proteger su información de un 99 por ciento de la gente que de con ella. La mayoría no serían capaces de resolverlo. Es más, ni siquiera tendrían una motivación lo suficientemente fuerte como para llevarlo a un experto criptógrafo. Así que, aunque su diario o sus transacciones económicas sean tan insignificantes como las mías, anímese a usar estos programas para ocultar información. Es como quedar oculto.

Si es extremadamente precavido, pase el mensaje cifrado cinco o seis veces por el programa y, cada vez, el mensaje se modifica más, como cuando pasa la misma masa varias veces por la máquina de hacer pasta. Por supuesto, para descifrarlo tendrá que ejecutar el programa de descifrado el mismo número de veces. Y, mientras lo haga, no use espacios entre las palabras; recuerde que es fácil leer texto sin espacios y, ¿por qué no introducir *nulls* aquí y allá como medida preventiva?

Nota: Si es todavía más que extremadamente precavido y quiere un sistema criptográfico cien por cien seguro, vea el programa one-time pad más adelante en este libro.

11.4. La superposición de Kerckhoffs

Con el paso del tiempo, la sustitución polialfabética, sigue siendo la más popular. Después de todo, es una técnica brillante y la base de gran parte de la criptografía actual.

Otro cambio importante en la naturaleza de los códigos secretos tuvo lugar con la invención del telégrafo. En períodos de guerra, permitía una transferencia frecuente, rápida y fiable de grandes cantidades de información cifrada (y casi siempre hay alguna guerra).

Auguste Kerckhoffs, holandés, escribió el texto que muchos expertos consideran el texto más significativo de la ciencia de la criptología. Publicado en 1881, *La Cryptographie*

Militaire era un folleto de 64 páginas que contenía maravillas. En él, señala el autor varios descubrimientos importantes. Primero, destaca que hay una diferencia fundamental entre los sistemas de cifrado que funcionan a baja escala (diarios, mensajes entre amigos) y las características, completamente diferentes, que requiere una transmisión masiva de mensajes durante una guerra (seguros, coherentes, no muy complejos de forma que puedan ser rápidamente cifrados y descifrados por soldados en el campo de batalla).

Otra observación significativa que hizo fue que la gente que diseña los sistemas criptológicos no son los que deberían afirmar el grado de seguridad del sistema. Para probar un sistema nuevo, hay que dárselo a un criptoanalista (un "resolvedor" de códigos secretos). Un criptoanalista puede probar objetivamente un sistema propuesto; especialmente sistemas cuyo uso es vital en guerras.

Sin embargo, el mayor triunfo del libro de Kerckhoffs es la explicación de su nueva técnica para resolver sustituciones polialfabéticas. Esta técnica se usa actualmente. Es la denominada superposición y requiere proporcionar al criptoanalista diferentes mensajes cifrados con la misma clave (generalmente esto es fácil de proporcionar dada la gran cantidad de mensajes cifrados que se generan en un día de guerra).

11.4.1. Construir una antitabla

Para usar el sistema de Kerckhoffs, hay que tomar diversos mensajes que compartan la misma clave y colocarlos formando una tabla donde los caracteres cifrados con la misma letra clave constituyan columnas. Por ejemplo, el sistema de la simple sustitución polialfabética tratado al principio de este capítulo requeriría simplemente que pusiese los mensajes secretos directamente uno encima del otro. La primera letra del primer mensaje es la primera letra de arriba en todos los mensajes subsiguientes. Creando una antitabla, puede intentar recomponer el original.

Para ver cómo funciona esto, usaremos primero el programa descrito anteriormente para cifrar los siguientes mensajes en inglés.

pleasedontforgetourcannons
pmgdwjjvvcpzdsieijvuij lr

thanksthefoodshipments
ticqoxlvzcrprbcsiyaigzjqj

theenemyisgettingstronger
tighrjsfqbpfgwcwjlkicbp

fivebomberswereheardlast
fjxhftsimachqeswurjwfoq

thecommanderwantsushome
tigfsrshvmocinbiilkaiha

A continuación, creamos nuestra antitabla colocando los mensajes cifrados.

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
p	m	g	d	w	j	j	v	v	c	p	z	d	t	s
i	c	q	o	x	l	v	z	c	r	p	r	b	c	s
i	g	h	r	j	s	f	q	b	q	p	f	g	w	c
f	j	x	h	f	t	s	i	m	a	c	h	q	e	s
t	i	g	f	s	r	s	h	v	m	o	c	i	n	b

Nuestro sistema polialfabético de este ejemplo es realmente sencillo; cada mensaje usa la misma clave (abcdefghijklmnopqrstuvwxyz), pero la táctica de desciframiento por superposición funciona también en sistemas complejos.

Cada carácter del texto cifrado se corresponde sólo a un carácter del mensaje original. Más adelante, todos los mensajes fueron cifrados con el mismo texto alfabético clave, por lo que los caracteres del mensaje original escondidos en una columna dada fueron cifrados por el mismo carácter clave. Para ser más específicos, cada texto original *p* escondido en la columna 1 se representará por el mismo carácter de texto cifrado, no cambia. Del mismo modo, cada texto *q* original está escondido detrás de un único carácter invariable, y así a través de todo el alfabeto. Como puede ver, la técnica de Kerckhoffs tiene el efecto de estabilizar los alfabetos cambiantes que hacen que la sustitución polialfabética parezca tan difícil de resolver.

Ahora que sus mensajes secretos han sido alineados (superpuestos), puede proceder a utilizar nuestra vieja técnica de análisis de frecuencias. Recuerde que cuando resuelve una sustitución simple, busque horizontalmente por todo el mensaje y cuente la frecuencia de los caracteres.

rax ax novsxx

Siguiendo con el ejemplo del inglés, tenemos que la letra más común, un trece por ciento, es la *e*, por lo que es muy probable que la *e* se corresponda con *x*.

11.4.2. Otros aspectos del polialfabeto

Para usar la superposición, cuente las frecuencias de los caracteres, pero en sentido vertical y no en horizontal. Trate cada columna como un código de sustitución ordinario (no polialfabético), lo que es en realidad.

Por ejemplo, en la tercera columna de nuestra antitabla encontramos esta repetición interesante

3
g
c
g
x
g

Podemos usar nuestro conocimiento de las estadísticas de frecuencia para dar rápidamente con el sistema de cifrado. Primero, estaríamos en el buen camino si aceptamos que cada *g* de la columna tres de este mensaje se puede traducir por *e*.

Podemos también aceptar que una *t* en la primera columna es siempre una *t*. La *t* es la segunda letra con mayor frecuencia de uso en inglés, y todavía más a comienzo de palabra. Es más, *the* es la palabra monosílaba más frecuente, con un 6,5 por ciento, y también la forma más común de comenzar un mensaje. Y, ¿no resulta curioso que el tercer y el quinto mensaje incluyan *tig*? ¿Se pregunta aún qué es lo que puede ser?

No cometeríamos ningún error si afirmásemos que el tercer y quinto mensajes comienzan por *the*. Esto significa que la *i* representa a la *h* en la segunda columna, y así sucesivamente.

¿Ha visto lo fácil que es descifrar el sistema polialfabético? Ahora es mucho más fácil de entender por qué Kerckhoffs insistió en que la gente que descifra mensajes son los más adecuados para juzgar su grado de dificultad.

Y no olvide que siempre hay palabras y sistemas que puede buscar cuando superponga mensajes en la vida real. Los comunicados enviados a la compañía aérea enemiga durante la Segunda Guerra Mundial seguro que incluían palabras como *Luftwaffe* (armamento aéreo), *Fallschirmjäger* (paracaidistas) o *Göring*, por ejemplo. O si se encuentra con un montón de mensajes cifrados en los bidones de basura que están detrás de las oficinas centrales de Revlon, le sugiero que piense en palabras como *barra de labios y colorete*.

Cuando se prueben los modelos de superposición, una de esas palabras aparecerá delante de sus entusiasmados ojos. Después, más partes de toda la antitabla empiezan a encajar a medida que se acerca a la tabla original del enemigo y a la solución del rompecabezas.

Fíjese también que, cuando descifra una palabra en un mensaje secreto, no sólo ha resuelto esa palabra, sino que también ha encontrado la solución a todo un tipo de modelos. Si descifra la palabra *the* en inglés, seguro que podrá descifrar también palabras como *there, they, then, he, hello, this, that*, etc.

A medida que la sustitución polialfabética se fue haciendo más compleja, varias técnicas de superposición surgieron como contraataque. El sistema St.-Cyr, simetría latente, proporción lineal, tablas esqueleto y otras tácticas con nombres igual de exóticos forman ya parte del arsenal del criptoanálisis.

Sin embargo, ninguna de ellas se acerca a la única y más importante herramienta para descifrar mensajes jamás inventada: el ordenador. Como veremos en el próximo capítulo, un ordenador puede probar incansablemente millones de antitablas. Un ordenador mueve las líneas y columnas de una tabla buscando, hora tras hora, modelos como *Luftwaffe* o *barra de labios* o cualquier palabra que crea que puede estar dentro del mensaje.

Un ordenador es como un hombre con millones de dedos, colocando las piezas de un rompecabezas de todas las formas posibles hasta que las piezas encajan. Cuando aparecieron los ordenadores, mucha gente dijo que ningún sistema criptográfico podría batir su búsqueda incansable. Lo que olvidaron es que, lo que una máquina descodifica también lo puede codificar. El juego se nos ha ido de las manos.

Esto no debería sorprenderle si recuerda que, en breve, el único jugador de ajedrez que podrá vencer a un ordenador será otro ordenador.

La llegada del ordenador

Alguna gente dice que la criptología es una parte de las matemáticas. No estoy de acuerdo. Es cierto que elementos matemáticos como la estadística, la teoría de la información, transformaciones de matrices y otros contribuyen a cifrar y descifrar sistemas. Pero la criptología es algo más que componentes matemáticos. Mucho más.

De la misma forma, algunos afirman que la programación informática es una parte de las matemáticas. Me parece también una simplificación excesiva. Gente poco formada en matemáticas ha demostrado ser un genio de la programación. De hecho, un estudio de los expedientes académicos de los mejores programadores demostró que había tantas personas de letras como de ciencias.

Resulta sorprendente que las principales contribuciones a la ciencia de la criptología no hayan sido realizadas por matemáticos. De hecho, figuras destacables como Alberti y Kerckhoffs eran generalistas, no matemáticos ni científicos en el sentido moderno de esos términos. Podrían describirse como aficionados, aunque el término ilustrados renacentistas es una etiqueta más respetuosa.

En cualquier caso, tanto la criptología como la programación resultan muy atractivas para iniciados que, a menudo, sobrepasan a los expertos en originalidad, ingenuidad y profundidad en su esbozo. Así que puede mantener vivas sus esperanzas aunque en sus estudios no escogiera la rama de matemáticas o informática. Va por buen camino. Si se ha informado y entretenido en los capítulos anteriores, los siguientes le resultarán igual de amenos e interesantes aunque, necesariamente, aumentan las referencias matemáticas y de programación.

12.1. Velocidad y precisión perfecta

Es obvio decir que los ordenadores causan un gran impacto en la disciplina de la criptología, que se está desarrollando a pasos agigantados. De hecho, ésta se desenvuelve tanto gracias a los ordenadores.

Dada la elevada velocidad y precisión perfecta con la que los ordenadores examinan y manipulan datos, las normas de la antigua criptología han cambiado por completo. Hoy en día no es suficiente con que el objetivo sea codificar información para que no la descifre el hombre: el objetivo de la criptología moderna es evitar que las máquinas consigan descifrarla.

Descifrar no requiere hoy en día que un inteligente criptoanalista se sienta a buscar pequeñas irregularidades y oscuros patrones. Por supuesto, acudir a la intuición y lógica humana puede funcionar a veces. Pero, contra las masivas transformaciones que puede realizar una máquina para cifrar mensajes, la única respuesta práctica es otra máquina para descifrarlos, es decir, un contraataque por ordenador que, incansablemente, escudriña el mensaje para resolverlo.

Interacciones de fuerza bruta interminables (denominadas *looping* en el lenguaje informático) pueden descodificar un mensaje sin ningún tipo de intervención humana. Por ejemplo, puede programar un ordenador para que manipule continuamente un mensaje cifrado de todas las formas posibles, con todos los reordenamientos posibles de los patrones y para que, al encontrar la palabra *the*, deje de buscar y le avise.

Los ordenadores son a la vez incansables e inconscientes de lo que están procesando pero, aún así, producen resultados asombrosos. Cualquiera que espere hoy en día ocultar información debe enfrentarse al hecho de que un ordenador "atacará" cualquier sistema de criptación en el que pueda pensar. Si quiere crear un código efectivo, debe retorcer y modificar el original de diversas maneras para evitar que el ordenador enemigo craquee el sistema usando su infatigable (pero, gracias a Dios, finita) búsqueda de patrones.

Los criptólogos calculan que si se dobla el número de permutaciones (formas personales de distorsionar el texto original) de un sistema de criptación, el criptoanálisis para hacerle frente debe elevar al cuadrado el número de intentos de desciframiento. Recuerde que a los ordenadores les en-

canta realizar miles de permutaciones por segundo. El cifrado actual de mensajes depende, a menudo, del creciente número de transformaciones hasta el punto que su descifrado llevaría miles de años de criptoanálisis.

12.2. Algunos puntos débiles en la criptografía informática

Desdichadamente, las técnicas criptográficas usadas anteriormente en aplicaciones informáticas como Microsoft Word se veían entorpecidas y debilitadas por algunos requisitos necesarios pero peligrosos. Por un lado, tenían que ser amables con el usuario, ser agradables para el usuario. Esto, desgraciadamente, les obligaba a incluir la contraseña (de alguna manera) dentro del fichero cifrado, lo que no era una buena idea. ¿Por qué lo hacían? (Word ya no posee esa característica de cifrado, que viene a ser lo mismo. Sin embargo, las primeras versiones de Word, hasta finales de los noventa, sí tenían esta deficiencia en su sistema.)

12.2.1. Contraseñas incluidas

Para que sea más cómodo para el usuario, los diseñadores de software incluyen un cuadro de texto en el que se le pide que introduzca la contraseña. Si ésta es incorrecta, el software contesta e indica que se debe volver a escribir. Esta pequeña cortesía afecta gravemente a la seguridad de un archivo cifrado. Después de todo, la mayoría de los criptoanalistas esperan recibir sólo un único mensaje codificado; no suelen tener el terrorífico *bonus* de que la contraseña vaya incluida en el mensaje. Al introducir la contraseña dentro del mensaje, consigue una interfaz de usuario bien diseñada (los usuarios se quedan desconcertados cuando algo no funciona y no reciben información). Sin embargo, esta cortesía tiene como resultado un cifrado defectuoso y vulnerable.

12.2.2. Demasiado fácil

Siendo justos, no se espera que los sistemas criptográficos de un procesador de textos, una hoja de cálculo u otras apli-

caciones comunes cumplan los estándares gubernamentales de alta seguridad. La adecuación, la accesibilidad al usuario y la velocidad son más importantes que una seguridad rígida. Sin embargo, hay personas que confían sus volátiles diarios personales a estos patrones. Del mismo modo, algunas empresas confían a la contraseña del procesador de textos su fórmula secreta de nuevas cremas faciales o sus planes para acabar con la campaña publicitaria de un rival. Si utiliza una aplicación de uso habitual que le ofrece cifrar sus archivos, yo le sugiero que rechace la oferta amablemente. La mayoría de ellos son fáciles de descifrar.

12.3. Cifrado informático básico

Recuerde que cuando un intruso intenta craquear su código, el primer trabajo del criptoanalista es buscar repeticiones. La repetición revela modelos, texturas. El intruso puede escoger buscar un número de términos o frases clave, entre ellos una contraseña insertada (que se encuentra en algún punto dentro del código cifrado) o, simplemente, patrones de la lengua natural.

Si encuentra un patrón de frecuencia en un mensaje, podrá llegar a descifrar un código de sustitución monoalfabético simple. Los ordenadores comienzan su ataque al mensaje cifrado de la misma forma que los humanos, sólo que ellos pueden hacerlo muchísimo más rápido.

12.4. Utilizar un código ya programado

Echemos un vistazo a un programa sencillo de Visual Basic que ilustra cómo utilizar un ordenador para cifrar un mensaje. En el siguiente sistema criptológico, hemos usado el código construido dentro del ordenador ANSI, que asigna un valor numérico a cada carácter del alfabeto. Los ordenadores almacenan caracteres de texto usando su propio código numérico ($a=97$, $b=98$, etc).

Este código de caracteres ANSI nos permite tratar cada carácter como si fuese un número, de forma que podemos hacer aritmética con esos caracteres. El siguiente ejemplo resta 1 al código numérico de cada carácter del mensaje original, distorsionando consecuentemente el mensaje y trans-

formándolo en un texto cifrado. Ponga un cuadro de texto en un formulario de Visual Basic y, a continuación, escriba el siguiente programa.

```
Private Sub Form_Load()  
Show  
a = "the message"  
For i = 1 To Len(a)  
    z = Mid(a, i, 1)  
    z = Asc(z)  
    z = z-1  
    Text1 = Text1 & Chr(z)  
Next i  
  
End Sub
```

Esta distorsión ha transformado el texto original "the message" en `sgd ldr`fd`.

Hasta el intruso más inepto puede descodificar este mensaje al darse cuenta de que hay tres *d* en el texto cifrado, que probablemente se corresponda a una *e*. Para facilitarle los patrones a nuestro intruso, hay patrones de frecuencia en todas las lenguas. Por ejemplo, como la palabra de tres letras más común en inglés es *the*, podemos deducir que esa es la primera palabra del mensaje. (Por supuesto, algunos sistemas de codificación llegan a extraer los espacios entre las palabras, pero si se encuentra *sgd* montones de veces en una cadena de caracteres codificados, seguramente habrá dado con *the* y, por extensión, con los caracteres *t, h, e*.)

Las tácticas empleadas en un programa informático para descifrar un mensaje son, como puede ver, precisamente las empleadas por el ser humano en el criptoanálisis. La única diferencia es que el programa puede analizar una mayor cantidad de texto y con más rapidez que cualquier persona.

A continuación, le mostramos un programa que analiza y muestra la frecuencia de caracteres de un texto cifrado.

```
Private Sub Form_Load()  
Show  
  
searchstring = "sgd ldr`fd"  
Text1 = searchstring  
  
l = Len(searchstring) ' get length of the  
ciphertext
```

```

cr = Chr(13) & Chr(10)

For i= 1 To 1

    searchchar = Mid(searchstring, i, 1) 'fetch
a character
    If InStr(donttry, searchchar) Then GoTo skipit
'already found

    Do
    p=p+1
        p = InStr(p, searchstring, searchchar)
        Counter = Counter + 1

    Loop Until p = 0

    Text1 = Text1 & cr & searchchar & " = " &
Counter -1

    donttry = donttry & searchchar

skipit:
Counter = 0
Next i

End Sub

```

Este programa produce el siguiente recuento de frecuencias y revela la posibilidad de que el carácter codificado *d* sea realmente una *e* en el texto original.

sgd ldr'fd

s=1

g=1

d=3

(space) = 1

l=1

r=2

'= 1

f=1

Fíjese que el mensaje es muy corto. Los mensajes en la vida real son mucho más largos y proporcionan, consecuentemente, mayores cifras de frecuencia.

Por ejemplo, cifrar el mensaje "the message when made longer reveals a better frequency count" tiene como resultado este texto:

sgd ldr'r'fd vgdm l'cd knmfdq qdud'kr àdssdq
eqdptdmbx bntms

Cuando ejecutamos el programa de análisis de frecuencias, obtenemos los siguientes resultados, más reveladores:

s=4

g=2

d=12

(space)=9

l=2

r=3

'=4

f=2

v=1

m=4

c=1

k=2

n=2

q=4

u=1

a=1

e=1

p=1

t=2

b=2

x=1

Hay un total de 61 caracteres en el texto cifrado y 12 de ellos (casi un 20%) son la letra *d*. El carácter cifrado *d* es probable que se corresponda con la letra del texto plano *e*. El carácter de espacio en blanco aparece 9 veces (y se corresponde al carácter de espacio en blanco).

Fíjese cómo este mensaje mucho más largo genera una carta de frecuencias con una mayor resolución. Cuanto más largo es el texto, más se adecuará a las estadísticas de frecuencia propias del idioma en el que ha sido escrito.

12.5. Un defecto fatal de Xor

Xor es una orden informática con una propiedad interesante: cambia los caracteres haciéndolos corresponder biyectivamente. Esto significa que cuando se introduce una *a* por primera vez, puede cambiarla a una *k*, y si se escribe una *k* puede cambiarla a *a*. Xor tiene una simetría fantástica.

Atraídos por esta simetría y otras características, muchos programadores informáticos utilizan Xor en sus sistemas de cifrado informático. Sin embargo, durante los 80 y 90 algunos programadores no eran conscientes de que si no se usa adecuadamente, Xor tiene un defecto fatal que veremos en el siguiente capítulo.

Infinitos monos: ataques por la fuerza bruta y otras curiosidades

Como ya sabían los faraones, la única forma perfecta de que la información permaneciese oculta era matar a los ingenieros, escribanos, canteros y otros trabajadores de las pirámides. Cortar la lengua a los trabajadores de las tumbas reales los desanimaría temporalmente, pero existía la posibilidad de dibujar croquis, aunque no supieran ni leer ni escribir. Así que la única solución efectiva era la muerte.

Durante miles de años, el ser humano intentó inventar el sistema criptográfico perfecto. Entonces algo cambió. Los trabajos de cifrado, descifrado y criptoanálisis empezaron a ser desarrollados por máquinas. Había llegado la era de los ordenadores.

No podemos cifrar un documento de forma que el mensaje codificado responda a un patrón independiente, pero podemos usar el poder de un ordenador para realizar tantas sustituciones y transposiciones que los patrones que existan en un texto cifrado queden efectivamente desfigurados y disfrazados.

Uso la palabra *efectividad* porque los métodos de cifrado por la fuerza bruta no postulan que sus mensajes no puedan ser solucionados jamás. De hecho, un método de este tipo no necesita ser a prueba de herramientas, ni siquiera en teoría, sino ser simplemente una solución que retrase el tiempo suficiente el descifrado del mensaje, de forma que cuando el enemigo lo recomponga, le sea ya inútil. La guerra ya ha acabado.

El enemigo intenta descifrar sus mensajes. Para que su sistema de cifrado sea efectivo, debe llevarle a su enemigo tanto tiempo descifrarlo que no le compense hacerlo. Por ejemplo, si le lleva veinte años al ordenador más rápido descifrar un mensaje enviado durante la operación Tormenta

del Desierto, el estatus de «mensaje clasificado» habrá terminado y la guerra será historia antes de que el ordenador del enemigo ofrezca soluciones.

Cuando los ordenadores juegan al ajedrez, no piensan de la misma forma en que lo hacemos las personas. Por el contrario, realizan un rápido análisis de grandes respuestas históricas y lo aplican a la partida que están jugando. Esta búsqueda es tan rápida y exhaustiva que pueden ganar incluso a los mejores jugadores humanos. De la misma forma, la criptografía informática también emplea lo que los humanos prefieren llamar métodos por la «fuerza bruta». El término «bruto» deriva de la historia de que si a un número suficiente de monos se les diese el tiempo necesario, podrían escribir todos los trabajos de Shakespeare (y los que habría escrito si no hubiese muerto).

Un número infinito de monos pueden producir cualquier resultado. De hecho, producen todos los resultados. Este es el pequeño fallo de este proceso: escribirían continuamente produciendo masas de texto y nadie tendría tiempo real para leer todos los libros escritos por los monos y comprobar dónde están las obras de Shakespeare. Sería una biblioteca del tamaño del universo, en la que la mayoría de los libros estarían llenos de palabras sin sentido, y, en alguna parte, estaría el *Hamlet* de Shakespeare. ¿Crear un catálogo? Desde luego, los monos también crearían universos de catálogos sin sentido. Alguno incluso podría decir dónde localizar *Hamlet*, pero nunca tendría tiempo para llevar a cabo la búsqueda. Ni siquiera los ordenadores pueden llevar a cabo búsquedas en universos de datos en un período de tiempo razonable. Hablaremos más de este modelo de infinitos monos más adelante en este libro.

13.1. Un problema con Xor

A pesar de los grandes éxitos obtenidos con los ordenadores en criptografía, ha habido también fracasos estrepitosos. Uno de los más interesantes es el sorprendente problema con Xor.

Hasta mediados de los 90, un considerable número de revistas informáticas publicaron artículos sobre la criptografía computacional y explicaban el valor de usar la operación binaria Xor, una de las órdenes disponibles en lenguajes computacionales.

Xor se usa mucho en criptografía computacional porque, entre otras cualidades, tiene la característica de conmutar cosas. Si aplica el operador Xor a un carácter una vez, este lo conmuta por otro. Si lo aplica de nuevo a este resultado, lo restaura a su valor original. De esta forma, Xor es, en realidad, una caja negra donde puede introducir un texto original y obtener un resultado aleatorio. Pero, si introduce de nuevo ese resultado aleatorio, le devolverá el texto original. Puede codificar y decodificar con la misma caja negra. La mayoría de los esquemas de cifrado trabajan de esta forma: codifica usando un proceso, por ejemplo, sustituir el símbolo # por cada e y decodifica invirtiendo este proceso.

Xor significa *O exclusiva* y funciona con dos números; en cierta forma, como la suma tradicional. Pero Xor es una operación que se efectúa a los niveles más bajos de la información: bits. Un bit puede tener sólo dos estados: 1 ó 0. Cuando introduce en Xor dos bits juntos, puede obtener cuatro posibles resultados:

$$0 \text{ XOR } 0 = 0$$

$$0 \text{ XOR } 1 = 1$$

$$1 \text{ XOR } 0 = 1$$

$$1 \text{ XOR } 1 = 0$$

Recuerde que en los ordenadores las letras del abecedario están ya en un código numérico simple, el código ANSI (compuesto de 8 bits por carácter).

Cada letra tiene un valor numérico (la A mayúscula es 65, B 66, etc.) En este nivel ya hay sustitución. Ahora, cuando aplique Xor a A junto con otro carácter, obtendrá otro número. Técnicamente la letra A se representa con un byte completo de memoria; un byte está compuesto de ocho bits, unidos en un grupo. Cuando introduce en Xor dos bytes, los dos conjuntos de ocho pares de bits se conmutan individualmente, obteniéndose así el resultado. Pero, ¿contra qué opera Xor un carácter introducido, por ejemplo, A?

Normalmente, Xor opera los caracteres del texto original enfrentándolos a una clave. A diferencia de un alfabeto convencional (abcdefg...) o de un grupo de alfabetos rotantes (sustitución polialfabética), una clave es una palabra o frase conocida por el emisor y el receptor, pero no para un intruso que intente descifrar el texto codificado. Puede imaginarse la clave como un alfabeto secreto y único.

Esta técnica añade, evidentemente, un nivel de complejidad a la labor del intruso. Nos ayuda a disminuir los patrones de frecuencia porque al usar una clave diferente se producirán modelos diferentes en el texto final codificado. Por ejemplo, si introducimos en Xor los caracteres *RM* bajo la clave *it*, obtenemos los símbolos ;9:

```
Private Sub Form_Load()
Show
origin = "R"
origin1 = "M"
Key = "i"
key1 = "t"
x= Asc(origin) Xor Asc(Key)
y = Asc(origin1) Xor Asc(key1)
Text1 = Chr(x) & Chr(y)
End Sub
```

El texto cifrado que obtenemos como resultado es:

;9

Si ejecutamos el texto cifrado de nuevo a través del sistema Xor con la misma clave, se recupera el texto fuente.

```
Private Sub Form_Load()
Show
origin = ";"
origin1 = "9"
Key = "i"
key1 = "t"
x= Asc(origin) Xor Asc(Key)
y = Asc(origin1) Xor Asc(key1)
Text1 = Chr(x) & Chr(y)
End Sub
```

El resultado es nuestro mensaje original, recuperado:

RM

La mayoría de los esquemas criptográficos basados en Xor toman del texto original una letra de cada vez. Utilizando Xor, enfrentan una letra a una clave, sea cual sea esta. Por supuesto, el emisor y el receptor del mensaje cifrado deben conocer la clave que descifra todo el documento. Normalmente, la clave es más corta que el texto que se va a cifrar.

En este caso, cuando se han usado todos los caracteres de la clave, empiece otra vez con la primera letra. Sin embargo, normalmente cuanto más larga sea la clave, mejor es el cifrado (tampoco es una buena idea una larga clave repetitiva como rrrrrrrrrrrrrrrrrrrrrrrr).

13.1.1. Un deslíz fatal

Xor parece bastante seguro, pero tiene un gran defecto del que nadie se percató durante años: si se utiliza 0 en Xor, no tiene lugar ningún cambio. El carácter no se cifra. Si ejecuta una clave con XOR contra una serie de ceros en el texto fuente, queda al descubierto la clave, el cifrado se ha resuelto. De hecho, en algunas situaciones en las que hay muchos ceros en el texto original, cualquiera que mire el texto cifrado verá la contraseña por todas partes, repitiéndose como los carteles de propaganda que se pegan en los muros. ¡Qué vergüenza!

Es poco inteligente pasar a través de Xor una clave escrita en una lengua natural igual al texto original. Este es el texto de un documento de Word cuando se abre con el Block de Notas. Esto es típico de la sección de cabecera de cualquier archivo .DOC. Los espacios en blanco son ceros.

?İà;±á > ?ÿ
 \ ! # ? ÿÿÿ ÿÿÿÿÿÿÿÿÿÿÿ

Ahora, veamos el mismo documento de Word tras el cifrado con Xor, usando la palabra ROAR como clave. Los ceros se han llenado con ella, de modo que todos pueden verla.

?İà;+áROARROARROARROAR>RO?ÿARROAR\ROARROARROARRO!
AR#ROARRO?ÿÿÿARROARROÿÿÿÿÿÿÿÿÿÿÿ

13.1.2. El cero numérico

Las revistas de prestigio no publican ya algoritmos Xor y no los usa ninguna aplicación que yo conozca. Es una forma estúpida de proteger documentos con claves (una revista técnica muy leída sobre Windows publicó el algoritmo en 1994, y fue la última vez que vi algo escrito sobre él).

No estoy diciendo que Xor sea inútil, sólo son inútiles malas implementaciones. Como verá en el capítulo siguiente.

te, Xor juega un papel importante en el famoso esquema DES (Data Encryption Standard) que ha estado sirviendo al mundo durante casi 30 años.

Quizás la razón por la que la gente usó los algoritmos Xor durante tanto tiempo es porque el texto puro no contiene ceros. Los ceros no sirven a ningún propósito a la hora de enviar un mensaje y no son parte del código alfabético ASCII o ANSI (el símbolo del dígito imprimible, que es el número 48 en el código ANSI, no es lo mismo que el *número* cero). Es más, el verdadero cero numérico no se puede introducir a través del teclado, sólo se puede escribir el dígito 0. Así que, hasta que aparecieron en escena otros archivos de procesadores de texto u otras aplicaciones relativamente avanzadas, simplemente no existían ceros verdaderos que se quisieran cifrar (excepto quizás un par de ellos al final de un documento, insertados por el ordenador para indicar el final de archivo).

De hecho, ¿quién tiene archivos con series de ceros en ellos? Prácticamente, cualquiera hoy en día. Los que propusieron la técnica simple de Xor no se dieron cuenta de que muchos archivos de imagen (.BMP, por ejemplo) contienen a menudo largas cadenas de ceros numéricos. Incluso archivos de texto creados con Word, WordPerfect u otros procesadores de texto contienen normalmente cadenas de ceros dentro de las zonas en las que no se forma texto (sin mencionar los documentos de texto y correo electrónico que cada vez más adjuntan imágenes al texto). Va a haber muchos de esos campos de ceros; campos donde las claves Xor pueden florecer de repente, como flores del desierto.

Si guarda un archivo .DOC que contenga sólo la palabra *hola*, ese archivo .DOC ocupará más de 19.000 bytes de su disco duro. Muchos de ellos son ceros, como se mostró en el ejemplo anterior.

13.2. Las limitaciones de las contraseñas

Algunas contraseñas se comparan simplemente con una versión oculta de ellas mismas y, si encajan, el mensaje se descifra rápidamente. Pero las contraseñas no sólo sirven de claves inmediatas. Este enfoque limitaría el papel de las contraseñas al mismo que desempeña la combinación de

una caja fuerte: si sabe la combinación, se abre la puerta y consigue el premio. Por el contrario, las letras de una contraseña se pueden usar para ayudar a distorsionar el mensaje original, como se ilustró con la técnica de Xor descrita anteriormente.

Aquí tiene un ejemplo diferente: supongamos que a cada letra del abecedario se le asigna un número ($a = 1, b = 2 \dots z = 26$) Ahora, use las letras de la contraseña en secuencia, sumando cada la letra al mensaje original. Cuando se quede sin letras en la contraseña, empiece de nuevo desde la primera letra. La palabra *the*, traducida a números que representen la posición de sus caracteres en el alfabeto, sería:

$$t = 20$$

$$h = 8$$

$$e = 5$$

Nuestro mensaje cifrado es 20 8 5

A continuación, usamos la contraseña *ba*, que se traduce en:

$$b = 2$$

$$a = 1$$

Al rotar la contraseña a través del original, sumamos cada par de letras del siguiente modo:

	20	8	5
+	2	1	2
	22	9	7

Obviamente, cuanto más larga es la contraseña, se repite menos durante el proceso de cifrado. Y, cuanto menos se repita, menos indicios habrá de un modelo y un análisis de frecuencias detectará menos repeticiones (al menos en base a la contraseña en sí).

Como la contraseña de nuestro ejemplo *ba* sólo tiene dos letras, la letra *e* en el mensaje original será codificada como 6 la mitad del tiempo y como 7 la otra mitad. Es probable que los números 6 y 7 sean los más comunes en este código (partiendo de que sea en lengua inglesa) y, si el texto cifrado es bastante largo, 6 y 7 tendrán el mismo nivel de frecuencia.

La siguiente letra más frecuente, la *t*, generará un par de números, 2 y 3, pero su frecuencia será inferior a 6 y 7. Estos hechos son bastante útiles al enemigo intruso persistente, muy inteligente aunque sin ningún tipo de ética. Recuerde que cuanto más largo sea el mensaje cifrado que cae en manos enemigas, más fáciles serán los cálculos de frecuencia. Si un análisis de frecuencia a lo largo del texto revela que los números 6 y 7 aparecen un 6,5 por ciento de las veces en el texto cifrado, tendremos que juntos representan un 13 por ciento, que es la frecuencia de la letra *e* en inglés. Es una buena pista que 6 y 7 representen a la *e*.

13.2.1. Ampliar la longitud de la contraseña

Las contraseñas mejores son aquellas que son lo más largas posibles y contienen dígitos y caracteres alfabéticos. Pero la mayoría de las contraseñas deben ser cortas por razones prácticas. Cuando le piden que introduzca una contraseña, normalmente usa el nombre de su perro u otra palabra sencilla. La gente no soporta tener que usar largas contraseñas o, simplemente, no son capaces de recordarlas.

Los números son más indeseables que las letras para las contraseñas. Los números están formados por diez dígitos y su nivel de frecuencia es plano; no puede esperar encontrar con más frecuencia un dígito o combinación de dígitos (823 es tan probable como 888). Con las palabras, sin embargo, tenemos esas combinaciones de letras, como por ejemplo en inglés *th*, que tiene un nivel de frecuencia muy elevado, frente a otras como *uu*, que en inglés sólo aparece en la palabra *vacuum* (vacío o aspiradora). Y la palabra en la que aparecen más vocales consecutivas es *queueing* (haciendo cola). A los criptoanalistas les encanta descubrir cosas de estas.

Desafortunadamente, la mayoría de las contraseñas usadas hoy en día (para cifrar archivos importantes, para acceder a un sistema informático o las BBS, etc) son palabras y no números. Algún software comienza a requerir ahora no sólo una contraseña de un mínimo de 10 caracteres, sino que también contenga, por lo menos, un número.

Las contraseñas utilizadas por un usuario común se componen generalmente de palabras porque un ciudadano medio puede recordar un máximo de siete números en un número de teléfono (y tiene problemas con siete). Los números PIN de las tarjetas bancarias suelen tener cuatro dígitos. A un ordenador le llevaría una fracción de segundo probar

las 10.000 combinaciones posibles de una contraseña de cuatro números. No es muy seguro el sistema.

Claramente, una buena solución al problema es escribir una contraseña tan larga como el propio mensaje. Esta es una forma de disminuir frecuencias que resultan al aplicar repetidamente la contraseña dentro del mensaje original (en contraste a esto, lo deseable es un texto original corto porque el campo desde donde se forman los patrones es más pequeño y estos no son tan obvios para el intruso).

13.2.2. Eliminar espacios

Muchos de los sistemas criptográficos que se usan en la actualidad (como los de los procesadores de texto) no permiten la eliminación de espacios entre las palabras. Al igual que la inclusión de la contraseña en alguna parte del texto cifrado, la inclusión de espacios es una cortesía para el usuario, al que no le gustaría demasiado que escribiera las palabras concatenadas en una cadena enorme cuando reconstruyera el texto. Por ejemplo, si quisieran imprimirlo, tendrían que ir por todo el documento restaurando esos espacios.

Como la media de la palabra en inglés es de seis letras de largo, el carácter de espacio en blanco pasa a ser el más frecuente, con un 14 por ciento de apariciones frente a un documento medio, comparado con un 13 por ciento que aparece la letra *e*. Si en un texto hay un carácter con una frecuencia de un 14 por ciento, puede afirmar que son espacios en blanco. Intente después sustituir ese carácter por espacios y obtendrá la longitud de las palabras. Esto es un gran avance si se quiere descifrar el mensaje.

DES: un sistema público

La criptología computacional arrancó a principios de los años veinte con la invención de una poderosa máquina. Durante eones, cifrar y descifrar fue trabajo de manos humanas, plumas y pergaminos, lápices y estilográficas. Pero cuando apareció la máquina Enigma, el trabajo pasó a ser función de la máquina y así será ya para siempre.

Construido originalmente para proporcionar seguridad a los empresarios alemanes, el aparato se convirtió más tarde en una herramienta importante en manos del ejército de Hitler. Es similar a un teclado encima de un gran amplificador estéreo, con todos los elementos dentro de una caja de madera. Si quiere ver unas fotos bonitas de Enigma, visite www.math.arizona.es/~dsl/ephtos.htm.

Sobre el teclado, hay un panel de luces que contiene el alfabeto. Si presiona una tecla, los tres rotores responden generando un carácter cifrado. Uno de los caracteres del panel se enciende y el operador escribe, a continuación, el carácter cifrado en un trozo de papel. Este proceso se repite hasta que se ha generado completamente el texto cifrado.

Una característica importante de esta máquina son los tres rotores. Están manualmente colocados en una posición pre-determinada anterior al cifrado y deben ser colocados en esa posición de cada vez. Los militares alemanes le fueron añadiendo rotores con el paso del tiempo, llegando hasta 12.

Al ser los primeros, los códigos típicos de tres y cinco rotores fueron resueltos por criptoanalistas británicos, muchas veces en sólo unas horas. Los británicos usaban análisis estadístico y otros métodos tradicionales de computación de frecuencias. Hacia el final de la guerra, los alemanes ya usaban el modelo de 12 rotores para transmitir comunicados entre los altos mandos. Los británicos habían construi-

do su propia máquina para descifrar los mensajes de Enigma, pero la baja velocidad de estos aparatos mecánicos vía telefónica provocaba un lento desciframiento.

Al final de la guerra, la maquinaria traqueteante fue sustituida por el calor silencioso de los primeros ordenadores de verdad, aparatos formados por infinidad de tubos como Colossus y, más tarde, ENIAC. Estas máquinas podían manipular información a una velocidad impresionante para la época: 25.000 bytes por segundo. Las ideas de los mandos supremos alemanes, incluyendo al propio Hitler, estaban ahora a disposición de los Aliados.

14.1. Para todo el mundo

Como todos sabemos, las inteligencias artificiales han ganado velocidad y potencia desde que se conectó la primera ENIAC. Desde la Segunda Guerra Mundial, se han estado llevando a cabo experimentos con las técnicas de la criptología, pero el mayor avance tuvo lugar a principios de los setenta. Se llama Data Encryption Standard (DES) y es, al mismo tiempo, extraña e intrigante. Por un lado, se hicieron públicos todos los detalles acerca de cómo trabajaba el procesador. Tradicionalmente, uno se guarda el secreto para sí mismo. La seguridad de un sistema de cifrado depende de no contar cómo funciona. Las dos partes que intervienen en el intercambio de un mensaje cifrado saben que se ha cambiado B por A, C por B, etc., pero nadie más conoce el proceso.

DES, por el contrario, funciona de una manera diferente, Todo el mundo conoce el proceso y, aún así, es casi imposible descifrar su procesamiento. Probablemente aún no se haya hecho, pero en teoría sí se puede.

De hecho, existe la posibilidad de que una agencia gubernamental haya descifrado el código de la máquina. No lo sabemos. Hubo movimientos sospechosamente extraños durante el desarrollo supersecreto del sistema DES.

Aunque la Agencia Nacional de Seguridad de Estados Unidos (NSA) estaba implicada en procesos de cifrado computacional, ni siquiera reconoció su existencia. Si le preguntaban a alguien que trabajase allí, negaban que trabajasen allí e, incluso, negaban la propia existencia de NSA.

Gran parte de la maquinaria y avances en la teoría de la criptología electrónica se enterraron en la burocracia más pro-

funda de los gobiernos norteamericanos. Entonces, de repente en 1972, surge otra agencia, la Oficina Nacional de Estándares (NBS), que aboga por la imposición de un único sistema criptológico estandarizado de Estados Unidos que podría ser útil tanto para las empresas como para el gobierno.

En un esfuerzo por salvaguardar la información automatizada, la NBS anunció que quería un método de cifrado computacional eficiente. NBS quería estandarizar la seguridad de datos (después de todo, es la oficina de estandarización). Querían algo fiable, totalmente seguro y barato en su implementación.

En realidad, ¿no es lo que han querido siempre a lo largo de la historia de la criptología los mandos supremos y los gobiernos? Pero NBS, sin hacer ruido y ejerciendo su influencia, consiguió lo que quería; casi siempre lo hacían. DES pasó a ser otro sistema que parecía infalible y, hasta ahora, sigue pareciéndolo. Aún así, se ha estado cerca de descifrarlo. DES, como veremos, usa 16 ciclos de cifrado y algunos criptoanalistas han conseguido descifrar hasta el número 15.

14.1.1. Lo que resulta extraño

La característica más curiosa de DES es que se hizo público su algoritmo (su forma de hacer el trabajo). Ese era uno de los requisitos del gobierno: todo el mundo podría ver cómo funcionaba.

IBM presentó un algoritmo de codificación denominado Lucifer que no es muy difícil de entender si se es programador: emplea una serie de transformaciones elementales en los bits del texto original y en la clave (usando operadores lógicos como XOR). Lucifer usaba muchas de esas transformaciones pero, una a una, resultan muy familiares y fáciles de entender. Es la combinación de muchas de esas transformaciones simples las que hacen de DES un sistema potente, de la misma forma que entrelazar fuertemente muchas cuerdas en una sola hace que esta sea lo suficientemente fuerte como para remolcar un barco de guerra hasta el astillero.

Originalmente, DES fue diseñado para usar una clave de 128 bits (16 caracteres), pero la Agencia Nacional de Seguridad que, como ya sabemos, no existe, insistió en que había que reducirla a menos de la mitad es decir, 7 caracteres. En su momento, hubo grandes críticas a que esta agencia estuviera implicada en el proyecto. Después de todo, para una organización que ni siquiera existía, estaba te-

niendo un gran impacto en los estándares y funcionamiento del nuevo DES.

Algunas críticas han sugerido que la NSA instaló una puerta trasera privada en DES, una puerta que sólo ellos y otras agencias inexistentes podían usar. A través de ella, NSA podría descifrar cualquier texto cifrado por DES. Otras sospechas apuntan a que se redujo el tamaño de la clave para que ellos pudieran descifrar el código.

A pesar de todos los escándalos, se adoptó DES con la clave más corta en 1976 y se convirtió en el primer estándar americano. Se publicó el algoritmo. Recuerde que jamás ningún gobierno o autoridad había hecho esto anteriormente; jamás se había reclamado una estandarización; jamás se había hecho público el funcionamiento de un buen sistema criptológico, ¿qué estaba sucediendo?

Al parecer, las relaciones entre NSA y NSB se habían roto. No había intención previa de que el sistema fuese de dominio público, pero se dio a conocer y la gente pudo usar esa información para crear software para DES.

Antes de que se pudieran pronunciar las palabras *institución secreta*, se había empezado a cifrar todo con DES, desde transferencias bancarias hasta correspondencia clandestina. En 1987, DES era ya el estándar a nivel mundial para instituciones financieras y de otros tipos; no tenía rival. Hoy en día, sigue siendo el estándar.

14.1.2. El funcionamiento de DES

DES realiza un verdadero trabajo de chef con los caracteres del mensaje. Utiliza sustitución y transposición, como la mayoría de las técnicas sofisticadas. En otras palabras, sustituye un carácter por otro y los mueve de forma que no conserven su posición original. Estas dos técnicas también reciben el nombre de *confusión y difusión*.

Si decidimos que nuestro cifrado sustituya *i* por *e* y *t* por *a*, *it* pasa a ser *ea*. Si queremos ir más allá y aplicar la transposición, podemos invertir el par de letras y que quede *ae*. Esta es la forma de combinar sustitución y transposición.

Recuerde que anteriormente decíamos que no importaba cuántas veces se realizara la sustitución y la transposición ni que reelaborara el mensaje con mucho esmero, ya que estas son las dos técnicas de las que históricamente disponen los criptoanalistas. Como veremos más adelante, el nuevo

sistema RSA utiliza otras transformaciones matemáticas más que sustitución o transformación.

Por supuesto, algunos argumentarán que la expansión, reducción y división en bloques son también técnicas fundamentales. En mi opinión no lo son.

En cualquier caso, puede concebir un sistema criptológico muy elaborado con la ayuda de un ordenador. Por ejemplo, puede decidir que su sistema ejecute 2.001 sustituciones y 4.351 transformaciones sobre el mensaje original. Los ordenadores facilitan las transformaciones a gran escala porque no se cansan ni cometen errores.

A los medio de comunicación les encantan las noticias del estilo: "Un error informático provoca que una mujer tenga que pagar un millón de pesetas de factura de teléfono". Seamos francos. El ordenador envió 20.000 facturas telefónicas correctas el mismo día. Es más, no es que en este caso el ordenador se hubiera olvidado de cómo se realizan las sumas. No fue el ordenador el que se equivocó, sino un ser humano que introdujo mal los datos. Si mete basura en el depósito de gasolina del coche, no se queje si este no arranca.

También con los ordenadores está limitado a utilizar esas dos técnicas fundamentales. No puede hacer más que esas transformaciones básicas y, además, ¿cuál es el propósito de estas? Intentar bloquear el posible uso de frecuencias, u otras técnicas, al criptoanálisis enemigo para que no encuentre modelos o patrones del texto original.

¿Sabe lo vulnerables que pueden llegar a ser las incansables técnicas de cifrado por la fuerza bruta ante las incansables técnicas de desciframiento por la fuerza bruta? Edgar Allan Poe, que creía que la criptología era fascinante, afirmaba que, dada la ingenuidad humana, era imposible crear un sistema para cifrar mensajes que no fuese descifrado posteriormente. Hasta el gran sistema Lucifer de DES está relativamente cerca de ser descifrado (en teoría, acabaría colapsándose, si es que no lo ha hecho ya).

Es demasiado pronto para afirmar que DES sobrevivirá para siempre. Históricamente, sólo hay un sistema que jamás ha sido burlado. Así que Poe casi tenía razón.

14.1.3. Los detalles técnicos

En las siguientes páginas nos sumergiremos en el funcionamiento interno de DES.

DES toma bits individuales del mensaje original y repite constantemente operaciones de sustitución y transposición rotatoria en ellos. El hecho de que DES funcione con bits en vez de con bytes (caracteres) expande considerablemente el "alfabeto" que se está usando, pero no introduce ninguna nueva técnica especial. Sin embargo, DES produce cadenas de mensaje cifrado que están cerca de ser totalmente aleatorios.

Primero, toma un grupo de 64 bits (8 caracteres) del texto original y usa la clave para transformarlo. Primero sustituye y luego transpone los caracteres del texto original contra la clave. Este bloque de 8 caracteres se subdivide, a continuación, en dos partes de 4 caracteres y se vuelven a realizar los procesos de sustitución y transposición 16 veces. A continuación, se vuelven a juntar las dos partes de 4 caracteres y se realiza un proceso final (invertido) de sustitución y transposición sobre el elemento de 8 caracteres. Fíjese que aunque ambos procesos se realicen 16 veces son siempre los mismos procesos; lo que se causa es un "amasado" del texto, como mezclar continuamente los ingredientes de un pastel provoca que cada vez se diluyan más.

Si le interesa, aquí tiene una descripción del proceso de DES a bajo nivel:

1. Se saca cada octavo bit en los ocho primeros caracteres de la clave. Esto reduce la clave de 64 bits a 56. A continuación, se divide la clave en dos mitades de 28 bits. Cada una de ellas se rota a la izquierda uno o dos bits, dependiendo de cual sea la iteración de 16 bits implicada en el proceso. Un desplazamiento hacia la izquierda de un bit tendría un resultado similar a este: abcd pasa a ser bcda. Por motivos ilustrativos, usamos caracteres completos. La rotación actual de bits funciona en base a bits, no en base a caracteres. En lenguaje C o Java, hay un comando para rotación hacia la izquierda (<<) pero, por ejemplo, no existe en Visual Basic.
2. Después del proceso de rotación, se extraen de la clave 8 bits más, dejando un total de 48 bits. El proceso de alterar el orden de los bits, junto con el proceso de omitir alguno, se denomina compresión por permutación.
3. Una expansión por permutación se realiza en la parte derecha de 4 caracteres del texto original. Se expande

de 32 a 58 bits. Ahora, la parte de texto original tiene el mismo tamaño que la parte de la clave y, consecuentemente, se puede ejecutar el comando Xor. La expansión es un proceso similar a la utilización de *nulls*, es decir, se hace el mensaje más largo de lo que realmente es, de acuerdo con un grupo predeterminado de reglas.

Se llevan a cabo sustituciones y transposiciones adicionales y, con los diversos pasos del sistema de DES, no supone ningún problema la cuestión de los ceros con Xor.

La fuerza principal y central del sistema DES son sus tablas de sustitución. Están diseñadas meticulosamente para producir resultados que sean difíciles de descifrar, con frecuencias planas esenciales. Hay ocho tablas y sus contenidos nunca cambian. Se mete un elemento de 6 bits y se obtiene uno de 4.

Los seis primeros bits de un texto plano se descomponen en dos números que le indican qué fila y qué columna de la primera tabla deberían usarse como resultado de texto cifrado. A continuación, se usa el segundo grupo de seis bits para saber que posición de la segunda tabla se debería usar y así sucesivamente.

Cada tabla tiene cuatro filas, numeradas 0, 1, 2, 3 y 16 columnas, numeradas de 0 a 15 (los ordenadores empiezan normalmente a contar desde 0).

Por ejemplo, supongamos que los seis primeros bits de nuestro mensaje son 010011. DES divide estos seis bits en dos números. Coge el primero y el último, que en este caso son 01, y debemos mirar en la fila 1 de la tabla. A continuación, usa los cuatro caracteres que quedan a la izquierda (1001) para indicarnos que miremos en la columna 9 de la tabla. De esta forma, encontraremos los valores de nuestro texto cifrado mirando en la fila 1, columna 9 de la tabla 1. Aquí tiene la tabla 1 de DES:

14	4	13	1	2	15	11	8	3	10	6	12	5	9	0	7
0	15	7	4	14	2	13	1	10	6	12	11	9	5	3	8
4	1	14	8	13	6	2	11	15	12	9	7	3	10	5	0
15	12	8	2	4	9	1	7	5	11	3	14	10	0	6	13

En nuestro ejemplo, introduciríamos nuestros seis bits (010011) y miraríamos en la fila 1, columna 9 para obtener

un resultado de 6. Si no obtiene este resultado es que no empieza la numeración de su fila y columna a partir de cero como hacen los ordenadores.

Como ya he dicho anteriormente, los contenidos reales de las ocho tablas nunca cambian. Muchos críticos se han preguntado quién ingenió estos valores numéricos y su disposición dentro de los cuadros. DES funciona muy bien pero los detalles de su creación han sido siempre una incógnita lo que, paralelamente, ha dado lugar a muchas teorías de conspiración, que se centran en los valores estáticos de las ocho tablas (¿qué es lo que llevan implícito esos valores determinados?; ¿quién los eligió y por qué?) y en la repentina reducción de la clave de 16 a 7 caracteres (si cuanto más larga es la clave, más seguro es el cifrado, ¿por qué acortarla?).

Además de la teoría de la puerta trasera, algunas críticas han indicado que quizá la NSA disminuyó la capacidad potencial de DES hasta un punto en el que ofreciera protección a los sistemas de comunicación a las empresas, pero que el propio gobierno pudiera descifrarlo con rapidez en caso de ser necesario.

A pesar de un pequeño estallido de censura, DES pasó a ser el estándar oficial de Estados Unidos en noviembre de 1976. El sistema DES se hizo público en enero de 1977, junto con la sugerencia gubernamental de que debería ser utilizada en cualquier transmisión de información federal que requiriese confidencialidad (bajo el nivel de seguridad nacional). El gobierno también animaba a las empresas a que utilizaran DES para sus propios intercambios de información confidenciales.

Los fabricantes de chips comenzaron a fabricar de los chips DES y el sistema se hizo popular. Aquí tiene una breve descripción que se hizo de DES en la Publicación de Estándares de Procesamiento, Información Federal, n° 81, con fecha del 2 de diciembre de 1980, titulado: "Especificaciones para los MODOS DE OPERACIÓN DE DES":

Los datos binarios pueden ser protegidos criptográficamente (cifrados), usando dispositivos que implementen un algoritmo especificado en el Data Encryption Standard (DES) (Publicación del FIPS, n° 46), junto con una clave criptográfica. La clave criptográfica controla el proceso de cifrado y debe usarse la misma clave en el proceso de desciframiento si se quieren obtener los datos originales. Al estar DES definido públicamente, la seguridad criptográfica depende de la confidencialidad de la clave criptográfica.

14.2. Desciframiento por la fuerza bruta

Al igual que los simples sistemas de Xor y otros modelos de cifrado, DES es simétrico. Funciona de la misma forma cuando descifra el texto cifrado que cuando cifra el texto original. Se introduce la clave y el texto cifrado; el mecanismo de DES lo mastica todo y acaba por escupir el mensaje original.

Si DES está disponible para todo el mundo (se puede comprar como chips de hardware o como software) y su clave sólo tiene 56 bits (siete caracteres, como la palabra *secreto*), ¿no sería una solución reorganizar esos siete caracteres de todas las formas posibles hasta dar con la clave correcta? ¿De cuántas formas se pueden disponer siete caracteres?

DES está disponible para cualquiera y los mensajes cifrados se obtienen con facilidad. Nadie teme que puedan ser descifrados, de forma que se utilizan ampliamente los mensajes cifrados con DES, incluso en Internet. La única cosa que protege a un mensaje de estas características es que la clave permanece en secreto. Así que, ¿quién iba a intentar probar todas las claves posibles? ¿No es esta la definición de un ataque por la fuerza bruta?

De hecho, sí. Pero hay un pequeño error en las ideas que se han planteado. La clave no se limita a la reorganización de los siete caracteres que la conforman, extraídos de un alfabeto de 26 caracteres. Si esto fuese así, DES sería inútil, ya que la fuerza bruta lo descifraría enseguida.

Recuerde que DES opera en el nivel de bits y que hay 56 bits en la clave de DES (cada bit tiene dos estados posibles, 1 y 0). Eso significa que el número posible de claves de DES equivale a 2 elevado a la potencia 56, lo cual da como resultado aproximado 7 seguido de 16 ceros.

¿Podría un ordenador chequear esa cantidad de claves? Seguro. De hecho, hay un 50 por ciento de probabilidades de que se resolviese cualquier mensaje secreto solamente probando la primera mitad de las claves.

En teoría un ordenador podría buscar todas las claves posibles de DES, pero este tendría que ser un superordenador carísimo. Hasta un superordenador Cray necesitaría miles de años para poder ejecutar las 70.000.000.000.000.000 de claves posibles con el texto cifrado.

Hasta ahora, no hay ningún método con el que enfrentarse a un texto cifrado con DES que no sea el de la fuerza bruta.

Pero recuerde que los ordenadores ganan potencia a un ritmo vertiginoso. ¿Quién conoce la capacidad actual de la maquinaria secreta del gobierno de los Estados Unidos?

También hay formas de interconectar ordenadores para que funcionen paralelamente, lo que le permitiría dividir la lista de claves de DES en partes más pequeñas. De vez en cuando, los aficionados han declarado que, juntando en paralelo sus ordenadores personales de esta manera, han conseguido descifrar textos de DES. Sin embargo nunca se han confirmado estos resultados.

Parece que no hay duda alguna de que, más tarde o más temprano, un ordenador tendrá la capacidad suficiente como para analizar los mensajes de DES y descifrarlos en un tiempo razonable. Una hora, por ejemplo. Y entonces será necesario adoptar otro estándar diferente, pero por ahora DES funciona de maravilla.

Claves conocidas

Hablemos de claves, tanto públicas como privadas. Recuerde que anteriormente se dijo que el esquema de cifrado de DES no ocultó ni su proceso (algoritmo) ni sus textos cifrados. Su éxito depende de que la clave sea secreta.

Sin embargo, hay sistemas modernos de cifrado que hacen pública más información y funcionan también bien. El proceso de cifrado, el texto cifrado y la clave; todo se hace de dominio público. Con toda esa información debería de ser capaz de resolver el texto cifrado, ¿verdad? Reflexione otra vez.

DES es un sistema importante y muy utilizado hasta el momento. Sin embargo, un sistema informático "rival", llamado RSA (de sus creadores, Profesores Rivest, Shamir y Adleman) utiliza un proceso de cifrado particularmente inteligente. RSA utiliza dos claves, una se hace pública y la otra se mantiene en secreto.

Esta es la primera vez en la historia de la criptografía en la que se usan dos claves: una para cifrar (la que se publica) y otra para descifrar (la que se mantiene en secreto). Hasta que este sistema RSA se empezó a diseñar a finales de los 70, nadie había pensado que la clave usada para descifrar pudiese ser otra que su inversa para cifrar.

Por ejemplo, siempre se había considerado fundamental que el primer paso en el proceso de cifrado implicaba mover, por ejemplo, el cuarto carácter al final del mensaje y luego el último paso en el proceso de descifrado debería ser mover el último carácter a la cuarta posición.

Otra cualidad particular del sistema RSA es que no utiliza ni la sustitución ni la permutación, los dos clásicos procesos de cifrado de mensajes, sino que echa mano de

manipulaciones puramente matemáticas de los caracteres, como veremos al final de este capítulo.

Cuando utiliza el sistema de clave pública de RSA, todos conocen la clave usada para cifrar el mensaje. Es como si permite publicar su número de teléfono en la guía, de forma que cualquiera pueda mirar y llamarle a casa. En la práctica, todos los que trabajan en su red tienen acceso a la lista de las claves públicas de cualquiera, pero esta lista no está disponible para cualquier extraño a la empresa. De todas formas, tampoco pasaría nada si un extraño se hiciese con estas claves.

La segunda clave, que es la que realmente descifra el mensaje, es privada y sólo la conoce la persona que recibe el mensaje. Con este sistema, miles de personas podrían emplear la misma clave pública (su clave pública) para cifrar y enviar mensajes; entonces utilizaría su clave secreta para descifrar todos estos mensajes. El esquema RSA también significa que no necesita intercambiar claves secretas con ninguna de esas miles de personas.

15.1. Solucionemos viejos problemas con las claves

Usar dos claves evita el viejo problema de los sistemas de una sola clave: tanto el emisor como el receptor del mensaje cifrado deben conocer la clave secreta, así que, ¿cómo se transmiten la clave entre ellos?

Podría cifrar la clave, pero esto sería una locura. Simplemente daría un paso atrás en el dilema porque necesitaría otra clave para descifrar la clave cifrada.

15.1.1. Llévela en un maletín

Una clave se envía a menudo a través de un guardia de seguridad (ya sabe: de estos que llevan un maletín de piel con cierre de seguridad esposado a sus muñecas). Pero este método es lento e inseguro. Y, podemos cuestionarnos, ¿quién tiene la llave metálica para abrir la cerradura? ¿Cómo se envió físicamente esa llave? La transferencia de claves es un viejo y complicado problema de la criptografía.

Hay otro problema importante con las claves. Puede suponer lo difícil que es mantener procesos comunicativos de

una sola clave entre muchas personas. Cada par de personas que envía un mensaje secreto debe tener una clave diferente (de lo contrario, podrían acceder a los mensajes ajenos).

El problema de múltiples claves es todavía peor de lo que pueda parecer, porque si trabaja en red, tendrá que generar un número de claves superior al número de trabajadores de dicha red. Debe hacer una clave para mí y para María, para mí y para Alberto, para mí y para Lucía, etc. Cada par de personas necesita una clave única. Por ejemplo, si hay 150 personas que se tienen que comunicar entre ellas, habrá que proporcionar 11.175 claves únicas. Es el número de posibles parejas en una red de trabajo de 150 personas.

15.1.2. Usar un centro de distribución de claves

Se han desarrollado distintas soluciones para tratar este problema de un exceso de claves. Una de ellas es un centro de distribución de claves en el que se guarda una clave DES para cada persona en un almacén central de seguridad. Si María quiere enviarle un mensaje secreto a Juan, se genera una "clave de sesión" temporal; después, la propia clave temporal se cifra usando la clave almacenada de María y se le envía a ella el resultado. Del mismo modo, la clave temporal se cifra usando la clave almacenada de Juan y el resultado se le envía a él. Cuando las claves temporales llegan a su destino, son descifradas por María y por Juan. Ahora ambos tienen la misma clave, que es la que se usa para cifrar y descifrar el mensaje.

Sin embargo, recuerde que si cifra múltiples mensajes utilizando la misma clave, aumenta considerablemente el riesgo de que un intruso sea capaz de descifrar sus mensajes. Usando un centro de distribución de claves se facilita la creación de una clave temporal para cada nuevo mensaje y evita dar al criptoanálisis la ventaja de tener múltiples mensajes cifrados con una única clave.

Otra ventaja obvia de la generación dinámica de una clave temporal única para cada sesión es que, aún cuando el intruso pudiese conseguir una clave, no supondría una gran amenaza para el sistema de seguridad. Sólo peligraría la información transmitida en esa sesión.

Las claves temporales también reciben el nombre de claves de sesión porque normalmente se utilizan en una única sesión y luego se destruyen. (Algunos sistemas generan una clave de sesión que se utiliza mientras María y Juan se en-

vían mensajes uno al otro, de forma que la clave se puede utilizar para más de un mensaje.)

15.2. La elegante solución RSA

Probablemente, la solución más elegante al problema de la distribución de claves sea el llamado método de la clave pública. Si utiliza el método de la clave pública de RSA, no se tiene que transmitir una clave secreta entre las dos partes ni se tiene que generar una clave única cada vez que haya un intercambio comunicativo. Por el contrario, Juan tiene su propia clave secreta y una clave pública que el resto de sus compañeros pueden usar para cifrar un mensaje y enviárselo a él. Del mismo modo, María le transmite a todos su clave pública y la secreta se la guarda. De esta forma, si la red de trabajo está formada por 150 personas, sólo hay que crear 300 claves, una pública y otra secreta para cada uno de los trabajadores.

15.2.1. Cifrados complejos

Como ya se puede imaginar, las claves pública y secreta funcionan conjuntamente para descifrar el mensaje. Es como abrir su caja de seguridad introduciendo su clave al mismo tiempo que un contable introduce la clave del banco. De hecho, en cuanto el emisor pase el mensaje por la clave pública, este alcanza un elevado grado de cifrado.

Si pierde el texto original, no hay forma de que el emisor pueda manipular el mensaje cifrado. Sólo la persona que tiene la clave correspondiente puede descifrar el mensaje.

El sistema de la clave pública de RSA funciona porque algunos tipos de operaciones matemáticas son fáciles de ejecutar en una dirección, pero imposibles en la contraria.

Veamos un ejemplo. Supongamos que le digo que el número 3 es mi clave pública. También le informo de que mi clave privada está formada de un grupo de números que, sumados, son la clave pública. ¿Cuánto tiempo le llevará probar todas las posibilidades hasta dar con la clave correcta? No mucho tiempo, ya que sólo hay cuatro: $1 + 2$, $0 + 3$, $0 + 1 + 2$, $1 + 1 + 1$.

Pero es bastante más complicado dar con todas las combinaciones posibles cuando la clave pública es un número grande.

Supongamos que mi clave pública es 14204925234522223453. ¿Qué hace? ¿Qué cantidad de números sumados dan como resultado esta clave?. Es un poquito más difícil, ¿verdad? Hay tropecientas mil posibilidades de combinaciones numéricas.

15.2.2. Fantasmas emergentes

Las operaciones matemáticas unidireccionales se denominan, a veces, puertas traseras. Se llaman así por las falsas puertas que hay sobre el suelo de un escenario del que, por ejemplo, puede emerger lentamente el fantasma de Hamlet a través de la niebla. Para el actor que realiza este papel, esperando el momento de entrar bajo el escenario, el pequeño elevador, los goznes y el pasador hacen que le resulte obvio dónde está la puerta. Sin embargo, para el resto de los actores, la puerta apenas es perceptible ya que se confunde con el resto del suelo.

La idea del proceso de la falsa puerta se puede realizar de forma sencilla en una dirección pero no en la otra. Piense en una albóndiga: es fácil picar un bistec utilizando una picadora. Pero es imposible coger la albóndiga e intentar que vuelva a ser bistec. Otro nombre para el proceso de la falsa puerta es función unidireccional.

***Nota:** Nadie ha demostrado que exista algo que sea una auténtica falsa puerta matemática. La mayoría de los matemáticos creen que éstas existen y que son irreversibles (sólo serían posibles ataques por la fuerza bruta que llevarían una eternidad). Sin embargo, es probable que una semana de estas un genio matemático como Fermat anuncie su elegante y eficaz sistema de factorización para números enteros de muchas cifras que hará desaparecer el esquema de RSA de la clave pública y a sus imitadores. Por todo el mundo, la gente se dedicará a recoger las claves públicas de los demás y podrán leer sus mensajes privados. Sería tan terrible como aquel día de 1782 en el que casi toda la corte francesa vestía seda china defectuosa y, cuando el sol de mediodía alcanzó su cenit, se evaporaron las ropas de todos los cortesanos, que en ese momento se hallaban en los arrebatadores jardines de Versailles (es una broma).*

La puerta trasera usada por RSA opera con números primos, números primos muy grandes multiplicados entre sí.

Un número primo es, simplemente, un número que no se puede dividir de forma exacta más que entre sí mismo y, por supuesto, 1 (cualquier número entero puede dividirse entre sí mismo y entre 1).

Dividir de forma exacta significa que el resto es 0. Por ejemplo, no hay otros dos números, excepto 1 y 5, por el que se pueda dividir 5 de forma exacta. Se puede dividir 5 entre 3, pero el resultado es 1,6667, y ese .6667 le indica que la división no es exacta. Hubo un resto, por lo tanto 5 es un número primo.

El número 4 no es un número primo porque se puede dividir entre 2 y obtener de resto 0.

El número 9 no es un número primo porque se puede dividir entre 3 y obtener de resto 0.

Finalmente, el número 26 no es un número primo porque se puede dividir entre 13 y obtener de resto 0.

Aquí tiene una tabla con los primeros 243 números primos.

2 3 5 7 11 13 17 19 23 29 31 37 41 43 47 53 59 61 67 71 73 79 83
89 97 101 103 107 109 113 127 131 137 139 149 151 157 163 167
173 179 181 191 193 197 199 211 223 227 229 233 239 241 251 257
263 269 271 277 281 283 293 307 311 313 317 331 337 347 349 353
359 367 373 379 383 389 397 401 409 419 421 431 433 439 443 449
457 461 463 467 479 487 491 499 503 509 521 523 541 547 557 563
569 571 577 587 593 599 601 607 613 617 619 631 641 643 647 653
659 661 673 677 683 691 701 709 719 727 733 739 743 751 757 761
769 773 787 797 809 811 821 823 827 829 839 853 857 859 863 877
881 883 887 907 911 919 929 937 941 947 953 967 971 977 983 991
997 1009 1013 1019 1021 1031 1033 1039 1049 1051 1061 1063 1069
1087 1091 1093 1097 1103 1109 1117 1123 1129 1151 1153 1163 1171
1181 1187 1193 1201 1213 1217 1223 1229 1231 1237 1249 1259 1277
1279 1283 1289 1291 1297 1301 1303 1307 1319 1321 1327 1361 1367
1373 1381 1399 1409 1423 1427 1429 1433 1439 1447 1451 1453 1459
1471 1481 1483 1487 1489 1493 1499 1511 1523 1531 1543

Ninguno de estos números es divisible por otro de forma exacta. Si cree que existe un patrón en esta progresión de

números, bien afortunado que es. Si quiere echar un vistazo a los primeros 10.000 números primos, visite la siguiente página:

www.math.utath.edu/~alfeld/math/p10000.html

15.2.3. Los números primos no son como los demás

Cuando se multiplican dos números enteros, cada uno de ellos se denomina factor del número resultante. En la multiplicación $5 \cdot 6 = 30$, 5 y 6 son los factores de 30 (30 tiene otros factores, como 2 y 15 ó 3 y 10). Los números primos, pobres, no tienen factores, excepto 1 y a sí mismos, por supuesto.

Pero los números primos son muy útiles como puerta trasera; son fáciles de manipular en una dirección y es prácticamente imposible invertir ese proceso.

El sistema RSA de la clave pública depende del bien sabido hecho de que si multiplica dos grandes números primos entre sí, es muy difícil deducir qué números primos se multiplicaron si sólo se tiene el resultado.

Cuando los números implicados son pequeños, por supuesto, no es tan difícil averiguarlo. Pongamos por ejemplo el número 15. Con sólo echar una ojeada a la lista de los números primos, podemos ver que los factores, los números primos que multiplicados entre sí dan 15, tienen que ser 3 y 5.

Pero, cuando utiliza números muy grandes, es imposible averiguar los factores. Si multiplica dos números primos de 100 dígitos cada uno, obtiene como resultado un número de 200 dígitos. Una nuez difícil de romper.

El sistema RSA funciona porque cuando se multiplica un número primo por otro, el número resultante no puede ser el resultado de la multiplicación de otro par de números primos distintos. Por lo tanto, sólo hay un posible par de números primos que, al ser multiplicados, dan como resultado un número concreto.

El resultado de la multiplicación de dos números primos es el producto de ese único par de primos. Es imposible, por lo que se sabe hasta ahora, factorizar el resultado (saber qué par de números primos fueron multiplicados para obtener ese resultado) cuando el número es demasiado grande. Así que puede hacer el resultado público (la clave pública) pero

sólo una persona conoce la clave secreta correcta (los dos números primos que se multiplicaron para dar como resultado la clave pública).

Dicho de otra manera, multiplicar dos números primos muy grandes es fácil. Sin embargo, es indecente la cantidad de tiempo que lleva factorizar esos números primos.

Las características de números primos multiplicados los hacen muy adecuados para crear pares de claves, una privada y otra pública.

Aquí tiene un ejemplo que muestra los detalles de cómo se construye un par de claves RSA:

1. El ordenador escoge dos números primos, cada uno de 100 dígitos (1.024 bits, por ejemplo) y los adecúa a ciertas normas (el ordenador quiere asegurarse, por ejemplo, de que ambas claves, después de ser manipuladas de diversas formas producen resultados de números enteros, no fraccionales). Para este ejemplo, sin embargo, ilustraremos el proceso con dos números primos pequeños: 3 y 19.
2. Multiplica los dos números. En nuestro ejemplo, $3 \times 19 = 57$. Este número se utiliza como la primera mitad de la clave pública.
3. El ordenador selecciona un número impar. Este número impar debe cumplir ciertas condiciones (no puede tener factores que sean números primos coincidentes con los dos números primos, menos 1, elegidos en el primer paso). Usaremos el 5 para nuestro ejemplo. Este número es la segunda parte de la clave pública. Así, ya tenemos nuestra clave pública: 57 5.
4. Se utiliza un simple proceso para formar la clave secreta.
5. Se le resta uno a cada uno de los números escogidos por el ordenador en los pasos anteriores: el primer número primo, el segundo y el número impar. Obtenemos como resultado 2, 18 y 4 y estos números se multiplican entre sí: $2 \times 18 \times 4$ y obtenemos 144. A continuación, se le suma 1 y obtenemos 145.
6. Por último, el 145 se divide entre el número impar escogido en el paso 3. Así que dividimos 145 entre 5 y nos da 29, que es la clave privada.

Así que el resultado final para la clave pública es: 57 5. Puede decirle a todo el mundo que le envíen mensajes cifrándolos con la clave 57 5.

Pero guarde ésta en secreto: su clave privada es 29. (Shhhhhhhhh).

Si viese una clave como 57 5 ¿cree que desde ahí podría llegar a la clave secreta? Buena suerte en su intento.

15.3. Es pura matemática

Como ya ha visto, la técnica de cifrado RSA utiliza un único sistema de claves. RSA es bastante inusual en otras formas. El sistema DES, manipula bits del mensaje original utilizando varias acciones de lógica booleana (como la rotación y XOR) y una combinación de técnicas criptológicas tradicionales como la sustitución.

Sin embargo, RSA manipula el texto original usando métodos puramente aritméticos, como la multiplicación, sobre cada letra del mensaje (no es necesaria ninguna sustitución o transposición).

Veamos un ejemplo de cómo se comporta un mensaje durante un proceso RSA. Es simple matemática. En RSA, cada letra del alfabeto (y cada dígito, signo de puntuación y símbolo) tiene un valor numérico simple (un entero positivo) asignado a él, empezando por $a = 1$, $b = 2$, etc. Puede usar el valor numérico de cada carácter para manipularlos matemáticamente.

Supongamos que el mensaje que quiere enviar es *hemos llegado*. Seguiremos el proceso de cifrar y descifrar la *h*, el primer carácter de su mensaje. Cuando se usa un sistema RSA, se repite el mismo proceso para cada carácter del texto original y, de esa forma, se cifra todo el mensaje. Sin embargo, por simplicidad, tomaremos sólo la *h* como ejemplo de cifrado.

Usaremos las claves obtenidas anteriormente. La clave pública: 57 5. La clave privada: 29. Aquí tiene lo que pasa:

1. La letra *h* tiene el valor 8 (es la 8ª letra del abecedario). Este valor se eleva a la potencia de la segunda mitad de la clave pública (5). Potenciar es multiplicar un número por sí mismo ese número de veces. El resultado de 8 elevado a la 5ª potencia es 32.768. Técnica-mente, este tipo de multiplicación se denomina *exponenciación*.
2. El siguiente paso es dividir ese 32.768 entre la primera mitad de la clave pública (57). El resultado es 574 con un resto de 50. (valor de cifrado: $32768 \text{ Mod } 57$).

3. Nos olvidamos del 574 y nos quedamos con el resto, 50. (Técnicamente, se denomina clase residual porque ignora el resultado principal de la división y utiliza sólo el resto. También se denomina aritmética modular).

La letra *h* ha sido cifrada. Se coloca en el texto cifrado como el primer valor del mismo: 50.

Ahora repita este mismo proceso con la segunda letra del texto original y continúe cifrando cada letra hasta que todo el mensaje haya sido transformado en texto cifrado.

Los pares de claves pública/privada tienen una relación simétrica (inversa) una con respecto a la otra. Por lo tanto, las operaciones matemáticas usadas para cifrar un mensaje utilizando una clave pública son las empleadas en el proceso inverso de ejecutar el texto cifrado a través de la clave privada y obtener así el mensaje original.

El descifrado sigue exactamente los mismos pasos matemáticos que el cifrado pero emplea la clave secreta en vez de la pública:

1. Eleve el valor del texto cifrado (50) a la potencia de la clave privada. 52 elevado a 29 da un número muy, muy largo.
2. El resultado del primer paso se divide a continuación entre la primera mitad de la clave pública (57) y el resultado tendrá de resto 8 (el valor de la *h*). De nuevo, nos olvidamos del resultado principal para quedarnos con el resto.

Valor del texto original = $50^{29} \bmod 57$.

Otra forma de describir el cifrado es esta función matemática:

$$E = (E \wedge P) \bmod \text{Pub } 1,$$

Donde *E* es el valor numérico del carácter del texto original, *P* el exponencial (segunda mitad de la clave pública) y *Pub 1* el módulo (primera mitad de la clave pública). El símbolo significa potenciación.

La función de descifrado sería así:

$$C = (C \wedge \text{Pri } 2) \bmod \text{Pub } 1$$

Donde *C* es el valor numérico del carácter del texto cifrado, *Pri 2* la segunda mitad de la clave privada y *Pub 1* la primera mitad de la clave pública.

Desafortunadamente, los procesos de cifrado y descifrado que utilizan estas transformaciones matemáticas son relativamente lentos. El sistema RSA necesita más tiempo para ser ejecutado en un ordenador que el DES. Como resultado de esto, el RSA es usado actualmente sólo en grandes redes informáticas donde se agudiza el problema de intercambio de claves. Sin embargo, a medida que los ordenadores aumentan su velocidad de procesamiento, es probable que DES sea fácil de atacar (quizás alguna agencia inexistente del gobierno haya conseguido *ya* descifrar sus códigos). En un futuro cercano, es probable que el sistema RSA pase a ser el sistema de cifrado preferido por aplicaciones extensas, desde el cifrado tradicional hasta la autenticación durante el proceso de transferencia de dinero vía Internet.

Recuerde que algunos sistemas utilizan hoy en día lo mejor de ambas técnicas: gestión de claves con el sistema RSA, pero codificación de mensajes con el sistema DES, más rápido. Y, si alguien inventa un sistema de cifrado basado en *cuantos* (descrito anteriormente), la apuesta queda abierta.

Firmas electrónicas

A veces chatea o envía mensajes de correo electrónico a gente que no conoce. No los conoce en absoluto, sólo sabe lo que ellos cuentan de sí mismos o lo que le dicen otros. Son ciber-amigos y colegas de trabajo. Probablemente ya conoce ese pequeño sentimiento inquietante; parecen buena gente, pero, ¿quiénes son realmente?

¿Cómo sabe si ellos son quienes dicen ser? De hecho, cuando enviamos y recibimos información electrónica, nos enfrentamos con la vieja paradoja de Shakespeare: ¿Es Shakespeare realmente otra persona, actuando como él y escribiendo sus obras? ¿Qué hay en un nombre? ¿Es importante? ¿Sería Hamlet menos interesante si fuese escrito realmente por alguien llamado Reina Isabel? No cambiaría ni una sola palabra.

Sin embargo, en muchas situaciones, la identidad es muy importante. Cuando va a tener una cita con alguien que conoció en un chat (generalmente una mala idea), va a comprar algo a un extraño en una subasta en Internet o va a autorizar a su banco que pague una factura, en muchas operaciones efectuadas a través de Internet se necesita autenticar la identidad de las personas.

Se sugieren muchos métodos actualmente que pueden asegurar que un mensaje proviene de un emisor determinado y nadie más. Por ejemplo, el sistema de cifrado de RSA puede hacer algo más que cifrar mensajes. También se puede usar para autenticar mensajes. RSA puede utilizarse para demostrar que un mensaje fue enviado por una persona más allá de toda duda. Esta capacidad tiene utilizaciones obvias en comercio electrónico, banca y otras situaciones donde la firma es una forma práctica de validar la identidad de al-

guien. Por ese motivo, se denomina en ciber-autenticación *firma digital*.

¿Ha considerado alguna vez los problemas inherentes a la autenticación informática? Cuando se trata de un contrato o un cheque en el mundo real, la autenticación implica casi siempre la presencia física del documento. ¿Ha borrado alguien algo? ¿Ha cambiado alguien el valor del cheque o falsificado la firma? A veces, con sólo mirar el documento, ya sabemos que alguien lo ha manipulado.

No es tan fácil alterar un documento oficial; hay que conseguir el papel adecuado o rescribir todo, falsificar firmas, etc. Pero, si alguna vez ha utilizado un procesador de texto, sabrá lo extremadamente fácil que son de manipular (cortar y pegar) los contenidos de los archivos del ordenador.

Incluso si alguien adjunta una imagen gráfica de su firma personal a un archivo, esto resulta inútil. Se puede capturar con facilidad esa imagen (presione la tecla ImprPant de su teclado) y, a continuación, péguela en cualquier programa de gráficos, como el accesorio Paint que viene con Windows. Recórtela (borre cualquier cosa que la rodee), y comience a escribir cheques por ordenador, pegándole la firma capturada. Los ordenadores facilitan mucho la falsificación de documentos. Por ejemplo, las copiadoras en color alcanzaron tal nivel de precisión en la reproducción de billetes de dólares en Estados Unidos, que estos tuvieron que ser rediseñados. Se le añadieron elementos para intentar evitar la falsificación: filigranas ocultas, tinta que cambia de negra a verde dependiendo el ángulo desde el que se mire, etc. ¿Qué es lo siguiente? ¿Hologramas?

16.1. Autenticación con RSA

Se han propuesto varios métodos que pueden verificar la autenticidad de archivos informáticos. Uno de los enfoques más sólidos es el método de la clave pública RSA.

Una firma RSA puede añadirse al mensaje utilizando las mismas técnicas que en el proceso de cifrado RSA. Esta firma verifica de forma efectiva la identidad del emisor.

Esta técnica asigna a cada carácter alfanumérico y símbolo su propio valor numérico ($a = 1$, $b = 2$, etc.). Una de las diferencias principales entre la firma RSA y el cifrado de mensajes RSA es que, al añadir su firma a un mensaje, el emisor usa su clave secreta. (Recuerde que durante el cifrado, el emisor utiliza la clave pública del receptor.).

A continuación, cuando el receptor recibe el mensaje, lo descifra usando la clave pública del emisor. Cualquiera que consiga este mensaje, por accidente o robo, puede descifrarlo también usando esa clave pública. Pero el propósito aquí no es cifrar. Simplemente se intenta certificar el origen del mensaje, marcarlo con una firma digital. No importa quién lo lea.

Cuando obtiene un mensaje firmado, si utiliza la clave pública del emisor y no consigue leerlo, significa que el mensaje no se ha descifrado porque el supuesto emisor no envió en realidad el mensaje.

De la misma forma, si alguien envía un mensaje firmado, no puede negar posteriormente haberlo enviado (es el único que puede hacer uso de su clave privada). Esto se llama la *no-repudiación*, como veremos más adelante.

Hay algunas variaciones a esta técnica. Por ejemplo, a veces se firma una pequeña verificación y después se adjunta al mensaje del texto original.

Si quiere firmar y cifrar un mensaje al mismo tiempo, firmelo primero con la clave secreta y luego cifre el resultado utilizando la clave pública del receptor. En el otro lado del proceso comunicativo, el receptor realiza el proceso inverso: primero lo descifra usando su clave privada y luego utiliza la clave pública del emisor para "desfirmarla". Es sólo después de este segundo paso cuando el texto original aparece y puede leerse.

16.1.1. La no repudiación

La característica de seguridad denominada *no repudiación* es similar a la autenticación y a la firma digital. Un aspecto importante de las comunicaciones en el ámbito legal y comercial es la habilidad para demostrar que ese emisor en particular fue, en efecto, el que originó la comunicación. La no repudiación también implica asegurar que en la comunicación no se produjo ninguna variación durante el proceso de envío. Esto es que nadie interceptó el mensaje o lo manipuló (por ejemplo, cambiar 400.000 Ptas. por 4.000.000 Ptas.).

Sin embargo, hablando estrictamente, la no repudiación se utiliza normalmente para referirse al hecho de que una persona no pueda seguir mintiendo cuando niega el envío de un mensaje concreto. Dicho de otra forma, con el software de no repudiación, se puede demostrar que alguien es el autor, por ejemplo, de una puja concreta por un cuadro vendido en EBAY.

16.2. El Congreso se ve implicado

En los Estados Unidos se aprobó una Ley para animar a los ciudadanos, empresas y el gobierno a usar firmas digitales de autenticación en sus transacciones de comercio electrónico. Por ejemplo, la ley permite que la gente utilice firmas electrónicas al cubrir solicitudes a través de Internet.

En junio de 2000, se aprobó la Ley del Comercio Digital del Milenio para dar al comercio electrónico un nuevo empuje. El Presidente Clinton firmó los documentos que otorgaban a las firmas electrónicas el mismo estatus legal que a las rúbricas tradicionales.

Como salvaguardas contra el abuso, se incluyeron algunas restricciones en la ley. Para proteger al 50 por ciento de los norteamericanos que todavía no disponen de Internet, la ley requiere que, por ejemplo, una empresa de servicios no puede cortar el suministro de luz o de agua comunicándolo a través de una notificación electrónica. Deben enviar la notificación en papel por medio de la Oficina de Correos. Extinciones de hipotecas, notificaciones de desahucio, órdenes judiciales, reclamaciones por producto defectuoso y otros documentos similares deben comunicarse siempre por papel.

De hecho, la ley dice que siempre se puede solicitar el uso de papeles si así se desea. Personalmente, me gustaría tener copias en papel de documentos como mi hipoteca y el seguro del coche, al menos hasta que se haya creado un *corpus* legal lo suficientemente fiable con respecto a esas firmas digitales y documentos a través de Internet. La ley permite que la compañía le cobre los gastos extra de envío si insiste en obtener los papeles. Parece algo injusto.

A menudo, la rapidez es el elemento esencial en varios tipos de comunicaciones y los servicios que ofrecen rapidez extra han conseguido satisfacer esta necesidad. Puede realizar contratos a través de FedEx (mensajería) o incluso por fax, por ejemplo y son una gran mejora con respecto al ritmo del correo tradicional. Resulta especialmente práctico poder firmar documentos electrónicamente, especialmente si hay varias personas que deben firmar el mismo contrato. Las comunicaciones electrónicas son funcionalmente instantáneas, lo cual es perfecto cuando su documento tiene que estar ahí ya, en ese preciso momento.

Recientemente he terminado de escribir un libro sobre como comprar un coche en Internet y descubrí que la mayo-

ría de los procesos de compra de coches en Internet era muy eficiente en la mayoría de los casos. Internet permite localizar rápidamente el coche adecuado con el precio ajustado, calcular los beneficios de alquilar frente a comprar, saber si el negocio merece la pena e, incluso, comprobar el historial de accidentes y reparaciones del coche usado que le interesa. Pero, cuando llega el momento de realizar la verdadera transacción, firmar los papeles, tiene que dejar a un lado el teclado y bajar hasta el concesionario. Los papeles serán el último elemento del comercio electrónico que se trate a través de Internet.

Como seguramente sabrá, el gobierno norteamericano es un gran consumidor de árboles, desde las solicitudes que hay que cubrir en IPO (Initial Public Offerings) hasta los proyectos de investigación de EPA (Environmental Protection Agency). El proceso de comunicación empresas/gobierno se está moviendo gradualmente del papel a Internet, pero la autenticación digital es un aspecto esencial en esta transición. Contratos, solicitudes, impuestos, informes; una gran masa de comunicación requiere mensajes legalmente firmados. Siglos de viejas técnicas como la rúbrica manual, el lacrado, los notarios, etc. se van quedando atrás. Todo el mundo se verá beneficiado del paso del papel a la electrónica, no sólo los árboles.

16.2.1. Surge la preocupación

El robo existe. A veces un dependiente o un empleado del servicio de basuras encuentra el recibo de una compra que realizó con su tarjeta de crédito. Obtienen su dirección y el número de su tarjeta. Lllaman y usan su identidad para comprar cientos de fantásticos equipos de música.

Alguna gente sigue insistiendo en coger el papel calco después de realizar un pago con tarjeta de crédito simplemente para prevenir este tipo de robos. Pero no importa lo caro que sea el equipo musical que el ladronzuelo compre, su crédito está limitado a 50.000 Ptas.

Desdichadamente, el mismo tipo de salvaguarda de crédito no está disponible aún en Internet para cualquier tipo de transacción. No forma parte de la Ley del Comercio Digital del Milenio u otra legislación similar recientemente aprobada. De hecho, si confía en la autenticación digital, no existe ningún límite a su responsabilidad. El cielo es el que marca el límite (esto no es verdad para compras con tarjeta de cré-

dito en Internet; está protegido por el límite de su tarjeta de crédito).

La Unión de Consumidores, que se encarga de la publicación de la revista Consumer's Reports (Informes para el consumidor), ha elaborado una lista con nueve consejos muy útiles para cualquiera que considere la posibilidad de usar firmas electrónicas para firmar un contrato. Puede encontrar estos buenos consejos en:

www.consumerunion.org/finance/digitaldc600.htm

16.2.2. El robo de identidades está a la orden del día

Aunque sepa muy poco sobre cómo manipular información y sobre servicios disponibles en Internet, podría robar fácilmente la identidad de alguien. Sin ningún tipo de esfuerzo, podría construir una serie de datos suficiente como para crear una identidad falsa, permitiéndole actuar como si fuese una persona real que compra cosas para "sus amigos" y que, en realidad, envía a la habitación de su hotel. Esto ha sucedido más veces de lo que admiten las empresas.

En los Estados Unidos, se puede conseguir la mayoría de los números de afiliación a la seguridad social (pruebe en <http://kadima.com>). Para robar la identidad de alguien en este país, necesita su número de la seguridad social, su dirección y el apellido de soltera de la madre. La dirección es fácil de conseguir y el apellido de la madre se publica en sitios como Who's Who, notificaciones de defunción, etc. Estos elementos de información son los que necesita para actuar como si fuese otra persona ante la empresa que acepte la tarjeta de crédito.

***Nota:** El robo de identidad suele tener lugar cuando alguien fallece. Se supone que los familiares deben cancelar las tarjetas en los bancos correspondientes. Pero a poca gente se le ocurre hacer esto en un momento de confusión y tristeza, fruto de la pérdida de un ser querido.*

Después de reunir la información necesaria, el ladrón llamará a la empresa o banco gestores de la tarjeta de crédito, se identificará como la persona a la que le ha robado la identidad y solicita un cambio de dirección fiscal. Supongamos que el ladrón está intentando pasar una semana o dos

en las suites del Hotel Ritz y recibir monedas de oro que él mismo enviará a través de FedExed a esa dirección. El ladrón le da el nombre de ese hotel a la empresa o banco de la tarjeta (el representante de atención al cliente no sabe que se trata de la dirección de un hotel). En cuanto a los números de teléfono, existen diversas formas de tratar el problema, como desviar la llamada, buzón de voz, etc. Para grandes compras, el ladrón puede hacer una transferencia de dinero desde la cuenta bancaria de la persona en cuestión a su propia cuenta de Visa. Enviar 20.000.000 de esta forma es menos problemático de lo que pueda parecer.

Si alguno de los lectores se siente ahora tentado, tenga cuidado. El robo de identidad es fácil de realizar pero muy difícil de continuar. Los métodos de detección mejoran constantemente y se producen cada día más detenciones.

16.2.3. Combinar RSA con DES

Como recordará de capítulos anteriores, los sistemas criptográficos de DES y RSA tienen diferentes ventajas. DES es más rápido y, por lo tanto, muy bueno para proteger información transmitida a alta velocidad. RSA parece ser más seguro, aunque nadie ha sido capaz todavía de descifrar el funcionamiento de DES. La característica de la clave pública de RSA hace al sistema especialmente atractivo para facilitar la comunicación entre los miembros de grandes equipos humanos. Sólo se necesita que cada persona tenga dos claves, pero el proceso comunicativo se puede llevar a cabo entre todas las combinaciones posibles de emisores y receptores dentro del grupo. Recuerde que RSA es también una buena forma de autenticar mensajes.

Sin embargo, es un sistema bastante lento debido a las fuertes transformaciones matemáticas que utiliza. De hecho, RSA puede llegar a ser de dos a cuatro órdenes más lento que DES. A veces, el sistema RSA no es en absoluto práctico para comunicaciones de alta velocidad o de gran tamaño. Es probable que, con el paso del tiempo, RSA sea cada vez más práctico a medida que va aumentando la capacidad de procesamiento de los ordenadores.

El punto más flaco de DES (como sucede con la mayoría de sistemas criptográficos con la excepción de RSA) es la necesidad de que dos personas intercambien una clave secreta antes de poder usar DES para descifrar el mensaje. ¿Se envía la clave por fax, a través de alguna empresa de seguri-

dad, por correo certificado? ¿Se susurra en el comedor o se llama por teléfono? Cualquiera de estos métodos puede potencialmente provocar que se rompa el sistema de seguridad. Lo peor, y esta es la parte difícil del problema, es que no hay forma de averiguar si esto ha sucedido o no. De forma que envía sus mensajes sin saber con seguridad si, valga la redundancia, están seguros.

Alguna mente privilegiada pensó: "¿Por qué no aunar lo mejor de los dos sistemas? ¿Por qué no usamos la característica de la clave pública de RSA para cifrar y transferir las claves que se usarán con los mensajes cifrados con DES?" Primero, distribuir las claves usando RSA y, posteriormente, intercambiar mensajes usando DES. RSA puede cifrar fácilmente una clave DES bastante rápido porque las claves DES son relativamente pequeñas (128 bits es lo normal y equivale a un mensaje de sólo 16 caracteres).

De hecho, este enfoque híbrido es exactamente lo que están haciendo en la actualidad muchas organizaciones. El siguiente capítulo ilustra cómo esta implementación híbrida DES/RSA ha sido utilizada en Windows 2000.

Implementaciones criptológicas en Windows 2000

Windows 2000 se jacta de poseer un sistema criptológico automático que es fácil de utilizar y altamente seguro. El *Encrypting File System* (EFS, Sistema de Cifrado de Archivos) cifra cada archivo con una clave aleatoria y emplea también las técnicas RSA y DES. Es eficiente y práctico porque una vez que se ha especificado el archivo o carpeta que se quiere cifrar, todos los procesos de cifrado y descifrado futuros son automáticos y transparentes para el usuario. Simplemente, trabaja con sus archivos como ha hecho siempre, pero si alguien intenta abrir, copiar, cambiar el nombre o mover los archivos cifrados, se activa un mensaje de error de **Acceso Denegado**. Y si intenta utilizar métodos de bajo nivel para intentar leer un archivo cifrado, sólo verán algo carente de sentido.

Antes de entrar en los detalles de cómo usar las herramientas de cifrado en Windows 2000, explicaremos un par de conceptos utilizados en el sistema de seguridad de Windows 2000.

17.1. Los aspectos básicos de SSL

Seguramente habrá oído hablar de SSL, *Secure Secret Layer* (Nivel de Seguridad). Desarrollada por Netscape, esta técnica inserta un capa de cifrado/descifrado (programa o grupo de programas que funcionan juntos) entre las aplicaciones y envía información dentro de una red y, por ejemplo, el nivel del protocolo de Internet TCP/IP. La técnica SSL se utiliza ampliamente en el envío de información importante a través de Internet, como por ejemplo, el envío del número de Visa en una transacción de comercio electrónico.

Como otros sistemas criptológicos actuales, SSL combina la tradicional técnica de la clave pública RSA con la clásica técnica de clave secreta tipificada por DES. Aquí tiene lo que sucede cuando se transfieren claves usando SSL y otros esquemas criptográficos conocidos.

1. El software de mi ordenador genera un número aleatorio de unos 128 bits de tamaño.
2. A continuación, ejecuto ese número a través de la clave pública para cifrarlo.
3. Le envío la clave cifrada.
4. Ejecuta la clave a través de su clave privada para descifrarlo. Recuerde que sólo usted conoce su clave privada y, consecuentemente, nadie más puede descifrar mensajes enviados usando su clave pública.
5. Ahora tiene la clave con un número aleatorio; le ha llegado la transmisión de forma segura.
6. Ejecuto mi mensaje a través de DES o un programa criptográfico similar utilizando la clave aleatoria que ambos conocemos.
7. Le envío el mensaje cifrado.
8. Ejecuta el mensaje a través de la clave aleatoria para descifrarlos, y ¡ahí está!

Las claves aleatorias utilizadas de esta forma reciben también el nombre de *claves de sesión*, porque tendré que generar una nueva clave la próxima vez que queramos comunicarnos. Para que este sistema sea seguro (contra las personas que puedan entrar y encontrar la clave cifrada aleatoria en su disco duro o en el mío) la clave sólo funciona durante esa sesión de comunicación. Cuando hemos terminado de hablar, se destruyen las claves aleatorias. Sin embargo, nunca se destruyen las claves pública y privada a no ser que alguien se haga con alguna de las claves privadas.

17.2. Certificados

Un certificado en este contexto es, como puede suponer, la confirmación de la identidad de alguien. Se utiliza a veces en Internet y lo verá en Windows 2000. Un certificado puede contener diversos datos identificativos, entre ellos una clave pública, una descripción del nombre del poseedor del certificado y otros detalles como el nivel de permiso de acceso a red (si puede, por ejemplo, editar archivos en ciertos directorios). Un

certificado puede usarse en Windows 2000 para identificar a alguien. Su clave pública es usada como el elemento visible del certificado. Windows 2000 también utiliza la técnica del certificado para verificar la autenticidad de los *driver* del hardware, de forma que le protege de virus u otras formas destructivas. (Un *driver de hardware* es el software que ayuda a un periférico, como por ejemplo el monitor, a hacer su trabajo.) Los fabricantes de hardware mejoran con frecuencia la capacidad de sus *driver* y le piden que descargue e instale el nuevo *driver*, por tanto lo tanto querrá saber si son los auténticos.

17.3. El sistema de cifrado de archivos de Windows 2000

Windows 2000 evita que los usuarios no acreditados (sin certificado) tengan acceso a los recursos del sistema del ordenador. Las propiedades del sistema de archivos (NTFS) y el permiso administrativo son relativamente seguros. Sin embargo, no hay motivo para ignorar el nivel extra de seguridad que puede ganar cifrando sus mensajes.

En las versiones anteriores de Windows, la seguridad a nivel de un archivo era bastante vulnerable. Podía activar un atributo de archivo denominado "protegido contra lectura" para intentar evitar que otras personas vieran el contenido de esos archivos. Sin embargo, la mayoría de los hacker pueden derrotar al sistema sin problema (utilizando el DOS).

Windows 2000 Professional ofrece una solución mucho mejor para ocultar su información. Y, además, es fácil de usar. De hecho, todo lo que tiene que hacer es activar la opción **Cifrar contenido para proteger datos** en el cuadro de diálogo **Propiedades** del fichero (o carpeta). Posteriormente, explico con más detalle los pasos que se deben seguir.

Puede seleccionar cifrar una carpeta con todos sus archivos o cifrar Certificados-Usuario actual cada uno de ellos por separado, aunque se recomienda cifrar carpetas y no sus archivos (las razones se explican más adelante en este capítulo).

17.3.1. Es automático y transparente

Sin embargo, el proceso de cifrado y descifrado pasa a ser automático y transparente al usuario una vez que se ha realizado el primer cifrado de una carpeta (o fichero).

Cada vez que abre un fichero cifrado, se ve el documento original, ya sea en Word o cualquier otra aplicación adecuada al fichero. Puede ver el documento original aunque en el disco duro resida como un fichero cifrado.

Haga doble clic en el nombre del fichero en el Explorador de Windows o bien use la opción **Archivo, Abrir** de una aplicación y el fichero se carga (descifrado) como si fuera un fichero normal. Puede copiar y cambiar el nombre a los ficheros que ha cifrado.

Excepto el propietario, nadie podrá entender lo que dicen los archivos cifrados. Ni siquiera podrán abrirlos o copiarlos. Obtendrán un mensaje de error "Acceso denegado" y su intento de apertura quedará grabado en el sistema. Nadie podrá editar ni cambiar el nombre de sus ficheros cifrados. Sin embargo, cualquier usuario con permiso para borrar sus ficheros también puede borrar sus ficheros cifrados.

Si alguien intenta leer el archivo mediante un editor de bajo nivel de DOS u otro sistema que lea byte a byte, lo único que verá será un amasijo de letras cifradas por el sistema DES de 128 bits.

Una vez que ha seleccionado la propiedad de cifrado, un fichero o todos los ficheros de una carpeta se cifran automáticamente al guardarlos y se descifran al abrirlos.

Parece haber poco o ningún ralentizamiento del sistema durante este proceso. Lleva el mismo tiempo guardar y abrir estos ficheros que los ficheros sin cifrar.

17.3.2. Puede copiar, los demás no

Puede cambiar el nombre de sus archivos a su gusto, sin preocuparse de que esto afecte al transparente sistema de cifrado/descifrado automático. Cada archivo tiene su propia clave única. Sin embargo, si copia un archivo sin cifrar dentro de una carpeta cifrada, el fichero no se cifra.

El nuevo Sistema de cifrado de archivos (EFS) de Windows 2000 no funciona con los sistemas de ficheros NT4 NTFS ni con los de Windows 95/98 FAT. Debe usar el sistema Windows 2000 NTFS (NT File System).

Tanto los usuarios particulares como los que trabajan en red pueden aprovechar la característica EFS para ocultar su información. Por defecto, cada vez que el administrador local (que es uno mismo en el caso de usuarios particulares) entra en el sistema, EFS genera automáticamente claves de recuperación y certificados autofirmados. Si es un usuario

particular, puede usar la utilidad de descifrado desde la línea de comandos de DOS para descifrar archivos. Sin embargo, el administrador de red puede variar el permiso para usar el agente de recuperación.

El proceso EFS comienza generando una clave privada para el usuario de forma automática y almacenándola junto con un certificado de su identidad. La asimetría ya conocida de la tecnología RSA (par de claves pública/privada) se utiliza para asegurar las claves. La técnica de simetría de DES se ocupa del cifrado y descifrado de los archivos. Técnicamente, la primera versión de Windows 2000 que salió al mercado empleaba el sistema DESX (En Norteamérica de 128 bits y en el resto del mundo de 40 bits). Sin embargo, se supone que las posteriores versiones de Windows tienen otro sistema de cifrado. De forma paralela, la primera versión de Windows EFS no tiene ninguna utilidad que le permita compartir archivos cifrados, pero se espera que las versiones posteriores permitan esto donde las personas de un mismo grupo puedan utilizar sus claves privadas para descifrar archivos.

Fíjese, sin embargo, que Windows 2000 EFS se limita a cifrar y descifrar archivos de su disco duro. Sólo oculta información almacenada en su máquina. No está diseñado para ofrecer una forma de transmitir un mensaje cifrado desde un punto a otro dentro de una red de trabajo o dentro de Internet, por ejemplo. Para eso, puede utilizar otras tecnologías como SSL.

17.3.3. Guardar archivos cifrados

Si quiere que sus archivos cifrados estén seguros, no se limite a copiarlos. Los archivos se copian como documento plano, sin cifrar. Por el contrario, acuda a la utilidad de Copia de seguridad u otro programa de copias de seguridad compatible con Windows 2000 EFS. Estos programas mantienen el cifrado durante los procesos de realización de la copia de seguridad y de recuperación de la misma, y no se necesita una clave privada.

***Nota:** Probablemente se haya dado cuenta de que muchas aplicaciones guardan archivos temporales. Los archivos con extensión .TMP son bastante frecuentes (búsquelos en las carpetas de Windows, Temp) Puede hacer que Word, por*

ejemplo, guarde automáticamente su trabajo cada cinco minutos por si hay algún corte de electricidad, por ejemplo. A veces, estos archivos no se borran automáticamente como se supone que debería ser. Y, mientras está trabajando con un archivo, su versión .TMP coexiste en el disco duro, esperando que alguien le eche un vistazo. Una versión .TMP no se cifra automáticamente, aunque sí lo hace el archivo principal. Por lo tanto, tiene que cifrar la carpeta completa allí donde haya almacenados archivos .TMP. Cifrar una carpeta automáticamente hace que todos los archivos contenidos en ella queden también cifrados.

Algunas precauciones de sentido común

Antes de que empiece a cifrar todos esos archivos comprometedores que tiene en el ordenador, debería tomar algunas precauciones. Tiene que ser capaz de descifrar los archivos en caso de que le pase algo a su clave privada y a su certificado (y todo puede pasar).

La semana pasada me quedé sin batería en el coche y, al fallar su fuente habitual de alimentación, mi radio "antirrobo" se apagó automáticamente, por lo que yo tenía que introducir un código para activarla. Esto ya me ha pasado otras veces. Es una verdadera lata. Los números de código que aparecen en el manual no funcionan. Este sistema se supone que tiene que dejar la radio en modo de apagado si alguien la roba. Lo que tuve que hacer fue ir al taller y allí sacaron la radio, localizaron el número de serie, llamaron al fabricante, se certificaron (con otro número secreto, por supuesto) y, finalmente, me dijeron el código para reactivar la radio.

Estoy seguro de que alguna vez le ha pasado algo parecido, tener que pagar un precio por los sistemas de seguridad tecnológicos que son necesarios en un mundo donde hay gente mala que hace cosas malas.

17.3.4. Cifrar un archivo

Los siguientes pasos le muestran lo fácil que es activar la opción automática de cifrar/descifrar archivos:

1. Inicie el Explorador de Windows (sobre el botón de **Inicio** de Windows, elija Programas, Explorador de

Windows o, para iniciarlo de una forma más rápida, presione la tecla de Windows del teclado y la letra E al mismo tiempo).

2. Localice el archivo que quiere cifrar.
3. Ponga el ratón sobre el fichero y haga clic en el botón derecho.
4. Escoja **Propiedades** del menú que se despliega. Verá el cuadro de diálogo que se muestra en la figura 17.1.

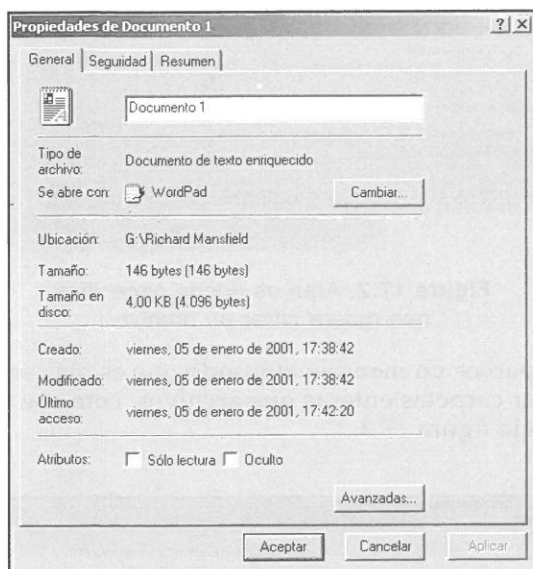


Figura 17.1. Todos los archivos despliegan este cuadro de diálogo de Propiedades

5. Haga clic sobre la pestaña **General** en este cuadro de diálogo.
6. Vaya al botón de **Avanzado**.
7. A continuación, verá el cuadro de diálogo de Atributos avanzados, como se muestra en la figura 17.2.
8. Active la opción de Cifrar contenido para proteger datos. (Si ha seleccionado la opción de comprimir, esta se desactivará porque no se pueden cifrar archivos comprimidos. Fíjese que Microsoft ha metido la pata

aquí. Una convención de Windows es usar botones excluyentes cuando una serie de opciones son mutuamente exclusivas. Sin embargo, aquí han usado incorrectamente las casillas de activación).

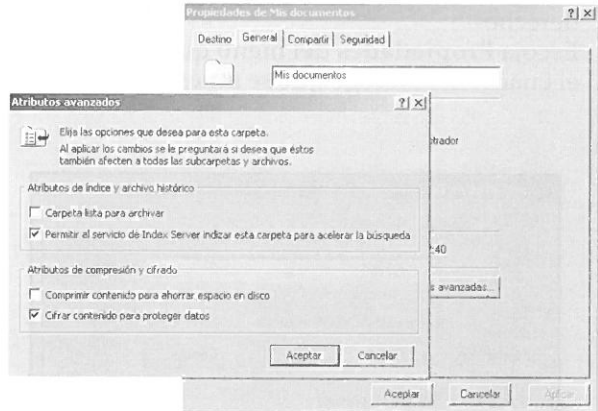


Figura 17.2. Aquí es donde especifica que quiere cifrar un archivo

9. Aparece un mensaje avisando que es más seguro cifrar carpetas enteras que archivos, como se muestra en la figura 17.3.

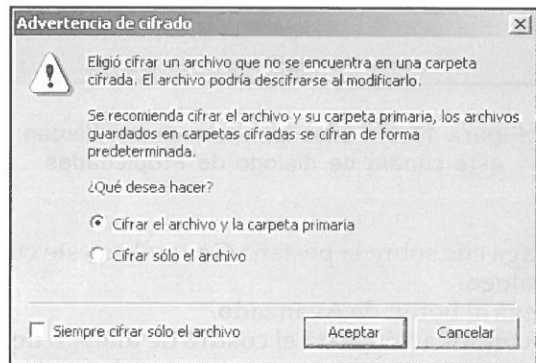


Figura 17.3. Puede elegir cifrar toda la carpeta en este cuadro de diálogo. Si cifra una carpeta entera, todos los archivos que guarde posteriormente en esa carpeta quedarán automáticamente cifrados

Nota: Si no ha podido seleccionar la opción que se cita en el paso 8, significa o que no está usando el sistema de archivos de Windows 2000 NTFS o que está intentando cifrar un archivo del sistema.

10. Siga el consejo del cuadro de diálogo de Advertencia de cifrado y escoja la opción de Cifrar el archivo y la carpeta primaria. Si no lo hace, la mera edición del archivo podría hacer que apareciese el texto original, anulando así el propósito de cifrado de todo el sistema. Recuerde: cifre carpetas, no archivos.

Nota: Si mueve o copia un archivo a un disco duro que no tenga el sistema NTFS, el archivo cifrado quedará descifrado y se almacenará una copia del texto plano en ese disco duro. Esto no es culpa de la magia, sino del diseño. El sistema EFS lo descifra antes de moverlo o almacenarlo. Sólo la persona que originalmente lo cifró puede copiar o mover el archivo. Si otra persona intenta hacer esto, se abre un cuadro de diálogo de acceso denegado.

17.4. Cifrar una carpeta

Para cifrar toda una carpeta, debe seguir básicamente los mismos pasos que para cifrar un archivo. Sin embargo, el cuadro de diálogo que aparece en el paso 9 es diferente (figura 17.3). Al cifrar una carpeta, se cifran automáticamente todos sus contenidos (los actuales y los que guarde ahí en el futuro). Sin embargo, puede desactivar esta opción. Puede especificar que quiere cifrar sólo la carpeta y, consecuentemente, cualquier archivo o subcarpeta que estén dentro de ella no será cifrado. Sin embargo, sí se cifrarán los que se añadan en un futuro.

Nota: Si normalmente archiva sus documentos en la carpeta Mis documentos, cifre toda la carpeta. Cifre también el archivo Windows\Temp o todos los archivos temporales que tenga almacenados en su ordenador.

Puede cifrar un documento con el sistema operativo DOS si quiere. Elija Inicio, Programas, MSDOS para abrir una

ventana de MSDOS. Escriba *cipher/?* y dele a Intro para ver las opciones disponibles en esta utilidad de DOS, como se muestra en la figura 17.4.

```

Servidor del sistema
Microsoft Windows 2000 [Versión 5.00.2195]
(C) Copyright 1985-1999 Microsoft Corp.

G:\>cipher/?
Muestra o modifica el cifrado de directorios (archivos)
en particiones NTFS.

CIPHER [/E | /D] [/S:dir-1 [/O] [/I] [/F] [/Q] [/H]
[/K] [nombre_ruta [...]]

/E          Cifra los directorios especificados.
            Los directorios serán
            marcados de forma que los
            archivos agregados serán
            cifrados.

/D          Descifra los directorios especificados.
            Los directorios
            serán marcados de forma que los
            archivos agregados no
            serán cifrados.

/S          Ejecuta la operación especificada en
            los archivos del
            directorio dado y en todos sus subdirectorios.

/A          Operación tanto para archivos como para directorios. El archivo ci-
            frado puede volverse a descifrar cuando se modifica en el directorio
            principal.
            Si no está cifrado, se recomienda que
            cifre el archivo y
            el directorio principal.

/I          Continúa la ejecución de la operación incluso después de
            producirse errores. De forma predeterminada, CIPHER se
            detiene cuando se produce
            un error.

/F          Fuerza la operación de cifrado en
            todos los objetos
  
```

Figura 17.4. Puede cifrar y descifrar desde la línea de comandos si lo desea

Las cachés, como la cola de impresión, suelen ser el objetivo de los hacker. Si deja sus archivos importantes dentro de un *buffer* como texto plano, contradice el propósito de cifrar esos archivos en otras localizaciones de su disco duro. Para solucionar este problema, intente descifrar todas las carpetas *buffer* (como la cola del estado de impresión) o, simplemente, no use ninguna cola de impresión.

17.5. Asegurar su clave y su certificado

El sistema Microsoft EFS de Windows 2000 no soluciona el viejo problema del cifrado de la clave secreta. ¿Se solucionará alguna vez? Probablemente, no mientras se usen claves.

Nota: Para proteger a principiantes, torpes y lentos, Microsoft ha construido una opción de guardar con contra-

seña en su sistema de cifrado EFS. Cuando cifra el primer archivo o carpeta, el sistema operativo almacena automáticamente una clave especial de recuperación que puede usarse posteriormente para descifrar la primera carpeta o archivo si perdiese su certificado o su clave privada.

Los certificados o su clave privada son gestionadas por el MMC (Microsoft Management Console). El MMC ofrece varios tipos de herramientas de administración que se dejan normalmente para que las gestione el administrador de red.

Sin embargo, debería a aprender a usar MMC para hacer una copia de seguridad de su certificado y de su clave privada (guardarlos en un disco y esconderlo en algún lugar seguro). Si alguna vez pierde su certificado y su clave (por ejemplo, un fallo del disco duro puede destruirlos), no podrá cifrar ni descifrar ningún archivo o carpeta cifrados. Pero, si ha tomado la precaución de guardar el certificado en un disco, puede recuperarlo junto con la clave y así, descifrar el documento.

Al tener esta copia de seguridad del certificado y de la clave le permite descifrar y gestionar sus archivos cifrados si, por cualquier circunstancia, tiene que llevarlo a otro ordenador. Si tiene que recuperar los archivos cifrados y llevarlos a una máquina nueva (supongamos que ha explotado su primer ordenador), debe cargar su certificado y su clave del disco de seguridad en la nueva máquina. Utilice MMC para importar su clave y su certificado (almacenados en el disco como un archivo .PDX, como se describe posteriormente).

Este proceso completo se explica en la sección "Importar un archivo .PDX con el certificado y la clave".

Siga estos pasos para guardar su certificado y clave privada en un disquete:

1. Abra MMC haciendo clic en **Inicio, Ejecutar** y, a continuación, escriba **mmc** y presione la tecla Intro.
Vea la figura 17.5.
2. De los menús MMC, como se muestra en la figura 17.5, escoja **Consola, Agregar o quitar complemento**.
Vea la figura 17.6.
3. Haga clic sobre el botón **Agregar**.
Vea la figura 17.7.
4. Haga doble clic sobre **Certificados** en el lista que aparece en 17.7. Vea la figura 17.8.

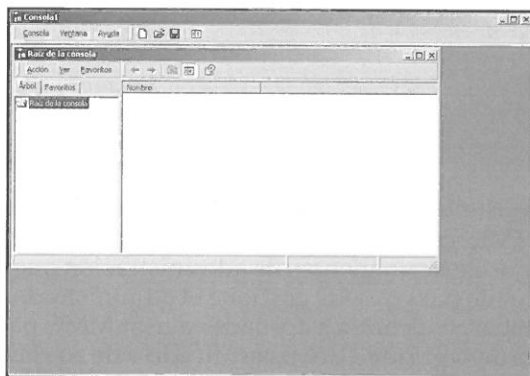


Figura 17.5. Aquí tiene la herramienta de administración del sistema MMC

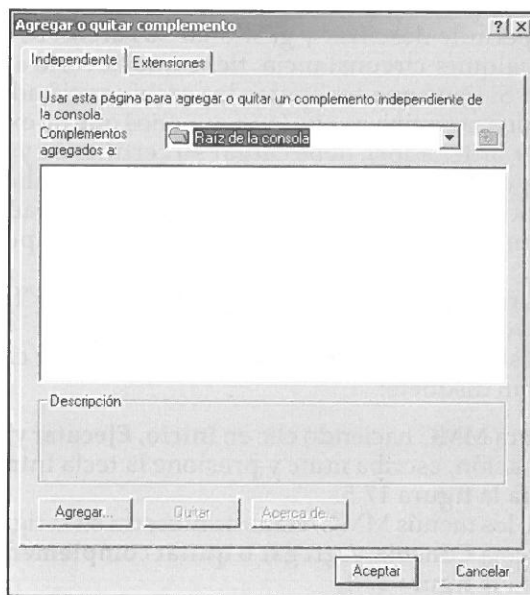


Figura 17.6. Los complementos son utilidades sencillas que puede agregar al MMC

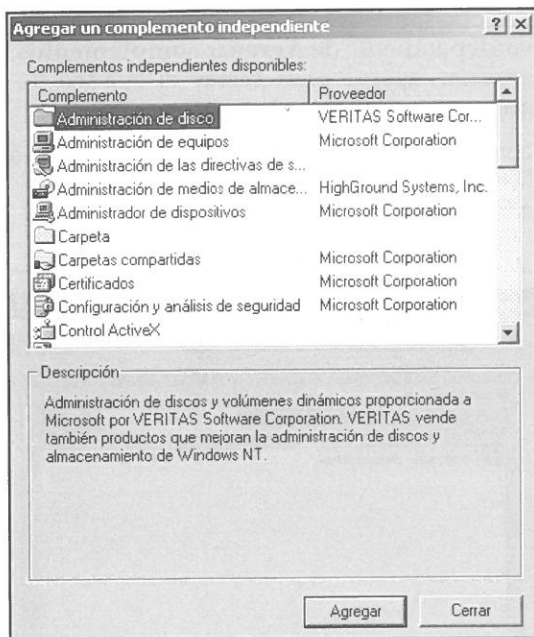


Figura 17.7. Aquí se listan todos los complementos posibles, incluido el que quiere: **Certificados**

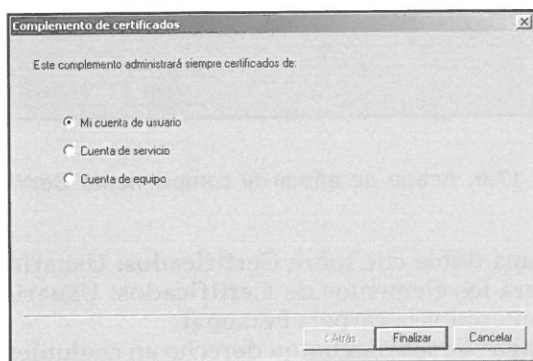


Figura 17.8. Debe seleccionar **Mi cuenta de usuario**

5. Deje activada por defecto la opción **Mi cuenta de usuario**.
6. Haga clic sobre **Finalizar**.

7. Haga clic sobre **Cerrar** para cerrar el cuadro de diálogo independiente de **Agregar complementos** y, a continuación, acepte para cerrar el cuadro de diálogo **Agregar o quitar complementos**.

Ahora verá que el complemento **Certificados** ha sido añadido a MMC.

Vea la figura 17.9.

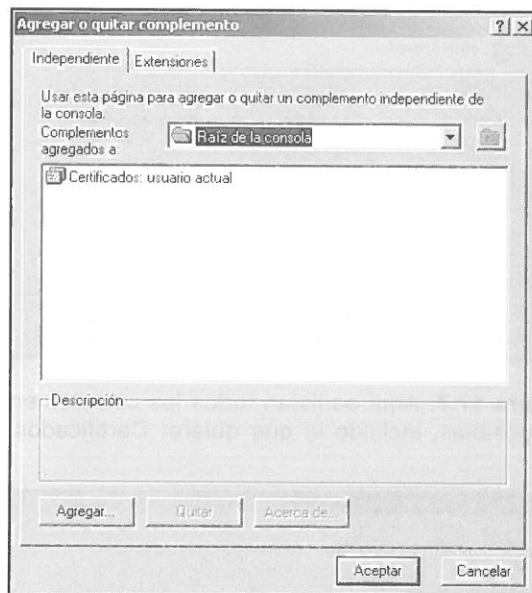


Figura 17.9. Acaba de añadir el complemento **Certificados**

8. Haga doble clic sobre **Certificados: Usuario actual**. Verá los elementos de **Certificados: Usuario actual** contenidos la carpeta **Personal**.
9. Haga clic sobre el botón derecho en cualquiera de los elementos que están bajo el encabezado **Personal** y escoja **Propiedades** del menú desplegable. Verá que el primer certificado es para recuperación de archivos y el segundo para EFS.
10. Elija **Acción, Todas las tareas, Exportar** (como se muestra en la figura 17.10).

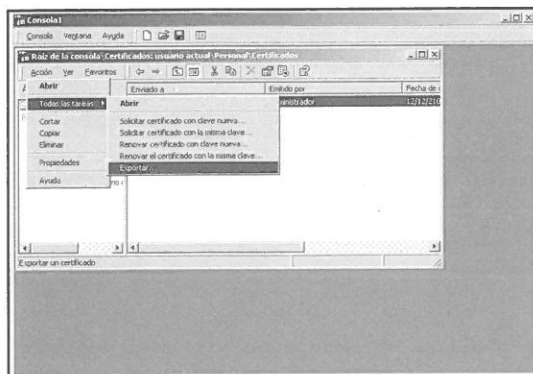


Figura 17.10. Guarde (exporte) su certificado y contraseña privada usando esta opción del menú

11. Cuando ha seleccionado la opción **Exportar**, se abrirá el **Asistente para Exportación de Certificados**, como se muestra en la figura 17.11.



Figura 17.11. Utilice este asistente para guardar su certificado y su contraseña

12. Haga clic sobre **Siguiente** en el **Asistente para Exportación** y le aparecerá el cuadro de la figura 17.12.

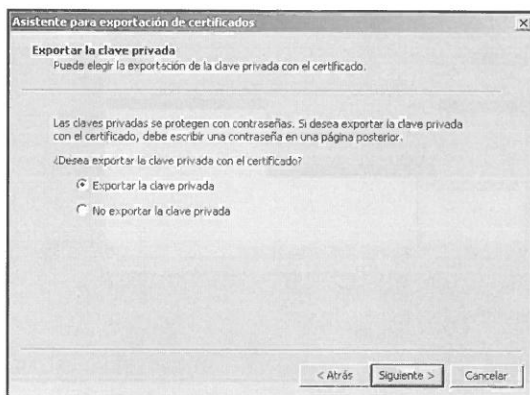


Figura 17.12. Deje esta opción por defecto si va a exportar su clave junto con su certificado

13. Deje activada la opción de exportación de clave y, a continuación, dele a **Siguiente** para ver las opciones de formato de archivo, como se muestra en la figura 17.13.

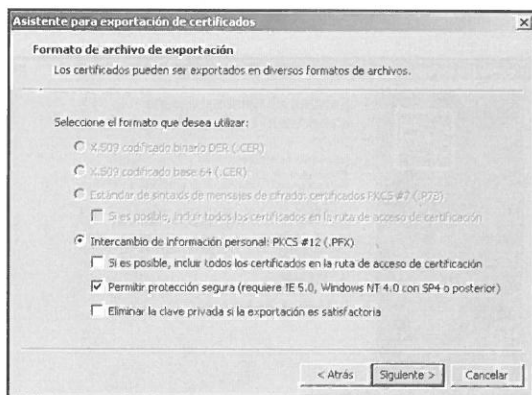


Figura 17.13. Deje la opción por defecto. Puede importar fácilmente un archivo .PFX

14. Deje activadas las opciones .PFX y permitir protección segura. A continuación, haga clic sobre **Siguiente** para ir la página **Contraseña**, como se muestra en la figura 17.14.

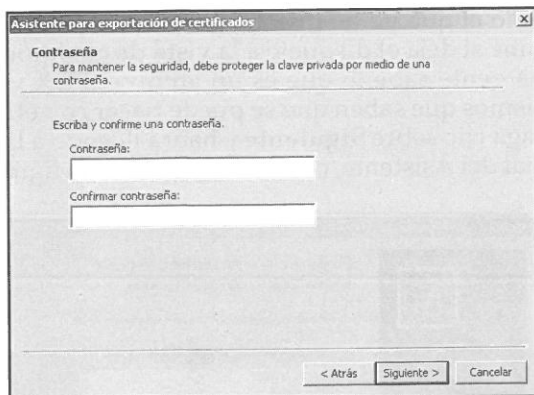


Figura 17.14. Proporcione una contraseña secreta en esta clave

15. Escriba la contraseña que prefiera, pero debería usar una contraseña segura (larga, evite palabras corrientes, incluya números). A continuación, haga clic en **Siguiente** para escoger ruta y nombre de archivo.

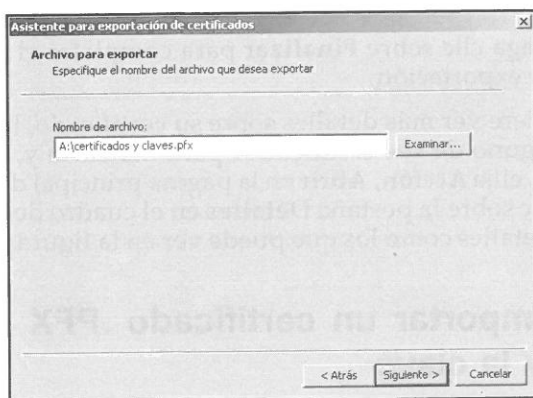


Figura 17.15. Escoja el nombre del archivo y qué unidad quiere utilizar para guardar su copia de seguridad del certificado

16. Seleccione la unidad y el nombre de archivo deseado. Puede que quiera utilizar uno que no sea tan evidente

como el que yo he puesto, pero no importa cómo se llame si deja el disquete a la vista de cualquiera. Mucha gente sabe lo que es un archivo .PDX y son los mismos que saben qué se puede hacer con él.

17. Haga clic sobre **Siguiente** y habrá llegado a la página final del Asistente, como se muestra en la figura 17.16.



Figura 17.16. Aquí se muestran los detalles del proceso de la exportación del certificado que acaba de completar

18. Haga clic sobre **Finalizar** para completar el proceso de exportación.

Si quiere ver más detalles sobre su certificado, haga clic sobre alguno de sus certificados para marcarlo y, a continuación, elija **Acción, Abrir** en la página principal de MMC. Haga clic sobre la pestaña **Detalles** en el cuadro de diálogo y verá detalles como los que puede ver en la figura 17.17.

17.6. Importar un certificado .PFX y la clave

El proceso de recuperar el certificado y la clave para su ordenador, o de importarlos en una nueva máquina, es bastante fácil. Siga estos pasos:

1. Arranque MMC y, si es necesario, agregue el complemento de certificados como se describe en la sección anterior.

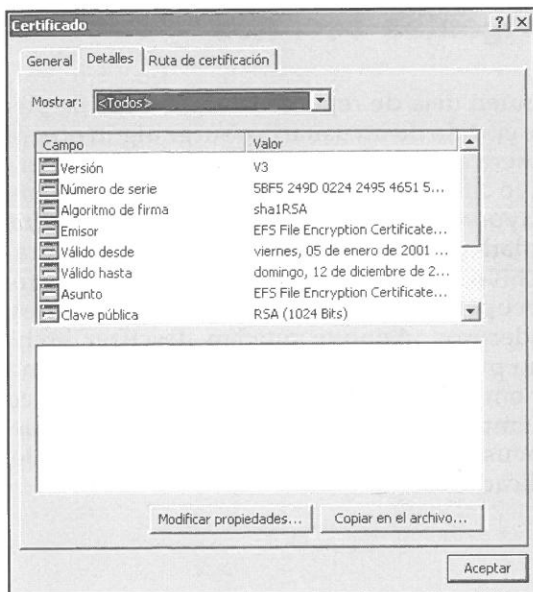


Figura 17.17. Aquí tiene los detalles. Fíjese que la clave pública en una clave RSA de 1024 bits. Esta sí que es una clave larga

2. Haga clic sobre la carpeta Personal, elija Todas las tareas y, a continuación, elija Importar. Seguidamente, verá el Asistente para importación de certificados.
3. Escoja la ruta (una carpeta) como, por ejemplo, c:\Windows donde guardó su archivo .PFX.
4. Escriba la contraseña que eligió en el paso 15 de la sección anterior.
Esto abre los datos del archivo .PFX.
5. Escoja la opción Colocar todos los certificados en el siguiente almacén. Haga clic sobre Acepto en el Almacén de certificados personales.
6. Haga clic sobre Siguiente y, a continuación, en Finalizar.
7. Acepte y comenzará la importación.

En este momento, se han instalado sus claves y ha sido certificado en la máquina. Ahora puede hacer con los archivos cifrados todo lo que hace normalmente con los que no lo están.

17.7. Agentes secretos

Si alguien deja de repente la empresa, o se pierden las claves de cifrado de un usuario o surge algún otro problema que requiera que el administrador tenga que tener acceso a los archivos cifrados, Windows 2000 tiene un sistema incluido para ayudar con esto. Se llama *Agente de recuperación* y es una utilidad que puede descubrir la clave cifradora de cualquier archivo cifrado (EFS genera automáticamente una clave de recuperación). Sin embargo, estos agentes no son superpoderosos. Aunque pueden descifrar archivos, un agente no puede descubrir la clave privada de un usuario. Es mejor omitir la descripción del proceso específico utilizado para emplear un agente de recuperación. Aquellos que necesiten usar el agente sabrán dónde encontrar los procesos implicados.

Esconder datos en corrientes de fotones

La mayoría de los teóricos creen que los sistemas actuales DES y RSA, como todos los sistemas criptográficos anteriores, deben pasar a la historia. Pronto serán craqueados porque los ordenadores son cada vez más rápidos. Incluso la impresionante técnica RSA sucumbirá algún día ante un ataque de un superordenador de alta velocidad.

Pero no se preocupe. Como siempre, en este antiguo juego de ping-pong entre criptógrafos y criptoanalistas, surgirá un nuevo sistema que todos creamos absolutamente indescifrable (hasta que alguien lo descifre, claro está).

Los criptólogos predicen que la próxima moda en criptología será la criptografía cuántica. Si se llega a poner en práctica, el sistema le dará un buen susto, créame. No ha habido nada similar hasta ahora. Y promete ser ya el último tanto, el final de la línea, la forma perfecta de cifrar información. Podría sacar de en medio a criptólogos y criptoanalistas de una vez por todas, de la misma forma que la imprenta dejó a copistas y monjes fuera de la actividad de copiar libros.

18.1. Criptografía atómica

A lo largo de este libro he intentado describir pasos prácticos y útiles que puede dar para protegerse de los diversos tipos de cibervandalismo.

Pero en este capítulo dejaremos el planeta tierra y exploraremos un poquito más. Consideremos la fantástica posibilidad de que en unos años se fabriquen dispositivos de cifrado submolecular. Mucha gente cree que la criptografía

atómica estará disponible más temprano que tarde. Mucha gente cree que el gobierno de los Estados Unidos ya ha construido este dispositivo.

Por lo que sabemos, los científicos trabajan todavía en el desarrollo de ordenadores de funcionamiento cuántico. Si llegan a construirse estas máquinas, serán muchísimo más potentes y rápidas que ningún otro ordenador actual. No estamos hablando de un progreso geométrico sino exponencial: una papelería de almacenamiento de 256 células que crece dentro de un contenedor tan grande que puede almacenar información sobre cualquier átomo del universo.

Por supuesto, los ordenadores de velocidad ultrarrápida se utilizarán para defenderse de los ataques por la fuerza bruta realizados a velocidad ultrarrápida sobre nuestros sistemas criptográficos. Hay tropecientos mil posibilidades de combinaciones de claves DES o RSA, y un ordenador cuántico podría ejecutarlas todas en un abrir y cerrar de ojos. Estas máquinas cuánticas llevarían la criptografía a un nivel completamente nuevo.

Como sabe, el mejor sistema de cifrado que tenemos hoy en día es RSA. Este aplica complejas transformaciones matemáticas sobre un mensaje y está concebido para ser extremadamente seguro. Sin embargo, es lento.

18.1.1. Cualquier combinación posible

Imagine, si puede, un ordenador cuántico que es capaz de descifrar un mensaje cifrado con RSA utilizando todas las combinaciones posibles de claves privadas en menos de un segundo. Estados de cuantos indistinguibles pueden amontonarse unos encima de otros de forma que abarquen cualquier variación posible simultáneamente. Esta cualidad de «todo al mismo tiempo», si se puede llegar a construir en una máquina real, destruirá obviamente los mejores sistemas actuales de cifrado (con la excepción del *One time pad* descrito en el capítulo 19). ¿Cómo podemos proteger información contra una máquina que puede chequear de forma inmediata cualquier clave posible?

La posibilidad de que se construya un ordenador cuántico (y quizá ya esté construido) ha mantenido a los criptoanalistas imaginando y diseñando constantemente posibles sistemas criptográficos cuánticos (QE). Hasta ahora los experimentos con QE han sido prometedores y se han obtenido varios éxitos en el mundo real. Es probable que un sis-

tema QE sea imposible de descifrar: por lo que se sabe hasta ahora, cualquiera que intentase descifrar un cifrado cuántico tendría que violar también los principios básicos de la física.

La propiedad de polarización de un fotón parece ser la forma más práctica de construir un ordenador cuántico, aunque también se está trabajando en la utilización del spin del núcleo atómico y del electrón como propiedades para transmitir información y ayudarnos a manipularla.

Las propiedades del spin o de la polarización pueden existir simultáneamente en dos estados al mismo tiempo, pero al combinarlos, se puede multiplicar el tamaño de la memoria o la capacidad de procesamiento sin que se vea afectada la velocidad ni haya un aumento de calor. Estamos hablando de un aparato muy denso y, virtualmente, libre de fricción. No queremos usar la expresión máquina de movimiento perpetuo, pero cuando nos ponemos al nivel de cuantos, aparecen un montón de cosas interesantes.

De lo que hay que darse cuenta cuando hablamos de máquinas cuánticas es que, hasta que se mide, cada electrón o fotón tiene una propiedad denominada *superposición*. Cada electrón gira en el sentido de las agujas del reloj y en el contrario simultáneamente o cada fotón tiene dos polaridades simultáneamente. Este don maravilloso le permite a un electrón o fotón realizar dos cálculos al mismo tiempo.

Si la balleta "ballerina" del anuncio pudiese retorcerse a izquierda y derecha al mismo tiempo, el público la vería fosforescente, incandescente. Y seguro que atraería a la audiencia.

18.1.2. Se enrarece la situación

Explicaremos cómo funciona todo. Imagine que pudiese hacerse muy pequeño y meterse dentro de una silla de roble para ver cada uno de los minúsculos elementos de esta (la pata, la madera, moléculas de la madera, átomos de las moléculas). Finalmente alcanza un punto justo por debajo del nivel de las moléculas, donde las leyes normales de la física cambian. Ahí la situación se enrarece mucho, demasiado.

Por ejemplo, intenta ver un electrón moviéndose en órbitas alrededor de un núcleo de un átomo de la madera de la pata de la silla. Pero no puede ver un único electrón girando alrededor del núcleo como hace la tierra alrededor del sol. El propio átomo no es un trocito de madera clara. Los atomistas griegos se equivocaron un poco.

Por el contrario, verá una esfera semitransparente, metálica y brillando tenuemente (una nube o un «escudo»), en todas las posiciones que un electrón pudiera tener alrededor de su núcleo. Parecería una burbuja bajo una potente luz solar.

A nivel de cuantos (nivel atómico) todo lo posible, aparentemente, sucede en todo momento. Esta característica de "todo al mismo tiempo" de la realidad cuántica resulta muy interesante para diseñadores informáticos y para criptólogos.

Dicha característica se corresponde al principio de incertidumbre de Heisenberg: no se pueden medir al mismo tiempo la posición y la velocidad de un electrón. Todo está pasando y su observación externa al proceso afecta a lo que está sucediendo. La medición causaría el colapso de la onda de probabilidad.

Los cuantos pueden existir en más de un estado al mismo tiempo. De hecho, cuando se agrupan, los cuantos pueden estar en infinitos estados, como infinitos son los puntos de una esfera.

Es como si la luna estuviese en todos los sitios posibles alrededor de la tierra en todo momento, formando una especie de carcasa esférica, gris, gruesa y grande en vez de la maravillosa luna que se desplaza en órbita linear, la que todos conocemos.

Ideas sobre el infinito

Probablemente haya escuchado esta frase hecha millones de veces: Las posibilidades son infinitas.

Esta expresión, o alguna otra variación, aparece con una habitualidad depresiva en los libros y revistas informáticos. Lo que los autores quieren decir realmente cuando usan esta frase es: "Los ordenadores son tan rápidos que pueden ejecutar cualquier permutación".

Pero el concepto de todas las permutaciones no es lo mismo que infinitud. Tenemos que tener cuidado cuando usamos la palabra infinito.

Los ordenadores actuales realizan las operaciones paso a paso, es decir, una detrás de la otra, aunque pueden llegar a ser muy rápidos. Sin embargo, están limitados a permutaciones y no pueden llegar a ningún punto cerca de una infinitud de pasos o estados.

Por ejemplo, el número de permutaciones de pares de parejas de baile cuando bailan cinco personas no es infinito. Es 20. Una forma de visualizar la verdadera infinidad es la siguiente. Imagine que le dicen que va a vivir para siempre. Como un favor público, decide pasarse el resto de su vida contando todos los números enteros 1...2...3...4 y así para siempre. Quiere demostrar que no hay final en la lista de números enteros, es decir, que después de haber contado 22222224444444 podrá contar 22222224444445. Infinito significa que puede contar para siempre. Nadie podrá aparecer e interrumpir su trabajo diciendo de repente "PARE, ya no hay más números. Ha alcanzado el límite. Ya los ha usado todos".

18.2. Mezcla de cuantos

Se ha demostrado que una propiedad de los fotones (partículas de luz) denominada polaridad es muy útil en el envío de mensajes, tanto a través de fibra óptica como del aire.

Los fotones producen un campo eléctrico que vibra. La dirección real de la vibración, denominada polaridad de los fotones, puede ser detectada. La polaridad puede estar en varios grados: 0, 45, 90 o 135.

Puede usarse la propiedad de la polaridad para transmitir mensajes. Puede enviar un par de fotones entrelazados moviéndose en todas direcciones (a gran velocidad, pues son fotones). Cuando se juntan, las dos partículas se comportan como si compartieran la misma identidad.

Algunos experimentos han demostrado que si dos partículas unidas se separan, si se cambia la polaridad de una de ellas, la otra partícula cambia su polaridad instantáneamente (¡más rápido que la velocidad de la luz! ¿Cómo pueden comunicarse?) Parece que hablan entre sí de una forma misteriosa, a través del espacio, más rápido que la luz. Esto abre caminos en Internet para mejoras impresionantes en el envío de información y también en la forma de computar de los ordenadores. Pero la velocidad no es la única cualidad atractiva del ordenador cuántico.

18.2.1. 250 Cubits = 1 Universo

Los cuantos también pueden transportar información, más información que la que contiene el propio universo. Cuando

se combina el extraño efecto del enlazado con el efecto también extraño de la superposición, se descubre que se puede procesar con una máquina cuántica o almacenar en memoria cuántica una cantidad de datos inconcebiblemente enorme. Si sólo se enlazan tres superposiciones, esta entidad puede albergar números entre 0 y 7. Ésta se denomina cubit (de *cuanto* y bit). Recuerde que un bit de memoria de ordenador convencional sólo puede operar con dos números, 0 y 1.

Un cubit puede llevar la misma cantidad de información que un byte (ocho bits) de memoria RAM tradicional. Ahora viene lo mejor: si se agrupan dos bytes se obtienen 16 bits. Pero, cuando agrupamos cubits, suman a mucha más velocidad. Cuando se juntan cubits la cantidad de información que pueden llegar a contener explota rápidamente.

Se ha calculado que un chip con 250 cubits enlazados puede llevar más información que el número total de átomos en el universo.

Otra forma de considerar esta característica maravillosa es que una máquina basada en cubits puede chequear todas las respuestas posibles a una pregunta de elevada complejidad instantáneamente (como por ejemplo el tiempo meteorológico de la próxima semana). Una máquina que puede mirar todos los estados posibles simultáneamente en un modelo complejo lleva la idea de la computación paralela más allá de nuestra comprensión actual. Seguro que no se le ocurriría jugar al ajedrez con ella.

18.2.2. Pero son tímidos

A pesar de lo potentes que son en ciertos aspectos, los estados de los *cuantos* son, por el contrario, frágiles y tímidos. Están sujetos al colapso instantáneo si algo de su medio los interrumpe de alguna manera. Esta gran sensibilidad también tiene sus aspectos prácticos. Además de su velocidad, de su densidad de datos y otras propiedades particulares, también son muy prácticos para detectar escuchas. En cuanto un intruso intenta mirarlos, se colapsan. Vuelven a su valor primario como un bit tradicional (se colapsa bien en 0 ó 1, en un sentido del giro o el otro).

Si un intruso intentase simplemente echar un vistazo a nuestra comunicación, nos daríamos cuenta inmediatamente (contendría un número inaceptable de errores cuando llegara a su destino, un 30 por ciento frente al 1,5 por ciento que podría ser generado por ruido, inestabilidad atmosférica, etc.).

Todavía quedan unos cuantos problemas por solucionar antes de que la criptografía cuántica pase al terreno práctico. Un largo desplazamiento a través de cables de fibra óptica, satélites o cualquier otro medio provocan en la actualidad que los estados cuánticos se mezclen o que los propios fotones se debiliten o sean absorbidos.

Aún así 60 kilómetros es una distancia lo suficientemente grande como para construir un cable de cifrado cuántico entre el Pentágono y el Oficinas Centrales de la Casa Blanca. No hay ninguna noticia acerca de que se esté usando un cable de estas características en Washington pero, ¿lo habrá?

Nota: Mucha gente no se da cuenta de que a pesar de lo que dijo Einstein de la velocidad de la luz (que era lo único absoluto en el universo relativo) sustancias comunes como el agua pueden reducir la velocidad de los fotones. Aunque los fotones pueden viajar sin problema a través de fibra óptica, al cabo de 50 ó 60 kilómetros, pueden absorberse fotones individuales. Einstein describió la velocidad de la luz en el vacío.

18.3. Dentro de la criptografía cuántica

Si combinamos el principio de incertidumbre de Heisenberg con la combinación cuántica se obtienen las herramientas para el mejor sistema criptográfico jamás concebido. Por lo menos, a este lado del paraíso.

Recuerde que la criptología tradicional tiene siempre un terrible punto débil. Alguien puede hacerse con los sistemas de cifrado/descifrado o con la clave.

De cualquiera de las dos maneras, el sistema queda comprometido.

Los sistemas tradicionales de cifrado siempre han requerido que dos personas que se quieren comunicar tengan que intercambiar primero una clave. El gran problema con este intercambio de claves es que alguien, el "interceptor", podría estar escondido tras la puerta del lavabo escuchando lo que se dice. O el guardia de seguridad que envía con la clave podría sentir curiosidad y echar un vistazo. O alguien podría interceptar un mensaje de correo electrónico. Las formas de comprometer los sistemas tradicionales criptológicos son múltiples. Y no hay forma posible de saber si algo ha

sucedido. Las personas que intervienen los mensajes no suelen dejar rastro de lo que hacen.

Así que debe seguir adelante y utilizar su sistema DES, con la incertidumbre de si alguien ha copiado su clave, o la de su receptor. No hay forma de saber si alguien ha intervenido el mensaje, ¿verdad? Incluso cuando se usa RSA, donde todos tienen una clave privada, ¿cómo saber si un intruso ha localizado una o más de esas claves y las ha copiado?

Como vimos anteriormente, incluso en la última versión de Microsoft Windows 2000, basada en un sistema RSA, se le requiere que guarde su clave en un disquete y, a continuación, lo esconda lo mejor que pueda. ¿Se cree que un espía empeñado en encontrarlo tendría algún problema? Todo esto no importa mucho si se trata de su cosas personales, pero sí cuando hablamos de gobiernos (a no ser que sea una Monica Lewinsky de la vida). Pero, para la información de seguridad nacional, o de seguridad de cualquier empresa, se requiere un sistema de seguridad que se base en algo más que en un disquete metido en el cajón de la oficina

¿Qué es lo que nos puede ofrecer la criptología cuántica? Se crea un par de fotones enlazados. Una máquina transmisora puede especificar una o dos polarizaciones de este par de fotones. Se trasmite uno de los fotones y el otro permanece con el emisor. Se compara esta polarización. El emisor y el receptor del mensaje miden la polarización.

Esto le permite detectar a cualquiera que intente intervenir en la transmisión ya que el principio de incertidumbre revela que el simple acto de observación de los fotones del par enlazado se separa. Es algo muy llamativo, como una alarma que se dispara.

Este efecto es similar al hecho de que cuando una persona se sabe observada, se comporta de forma diferente a cuando está completamente sola.

En el nivel de *cuantos*, el mero hecho de la observación es detectable.

De este modo, si se intercepta un mensaje que está siendo enviado, se sabrá que alguien más está escuchando el mensaje, por lo que se puede cerrar la transmisión de fotones e intentarlo más tarde.

18.3.1. Alice, Bob y Eve

Los criptólogos denominan por convención a las tres partes de la transmisión de un mensaje secreto Alice (emisor), Bob (receptor) y la mala Eve (el intruso).

Tendré más que decir acerca de la perfección del sistema *One time pad* en el próximo capítulo, pero lo utilizaré aquí brevemente. Es indescifrable, aunque tiene algunos aspectos de ineficiencia. Aquí tiene un breve resumen de cómo funciona.

Alice transforma el mensaje en bits (unos y ceros). Supongamos que quiere enviar el mensaje AB. Esto es fácil de hacer para un ordenador; funciona sólo con bits. La letra A se representa en el ordenador por el número 65 (B es el 66, etc.). Cualquier número puede transformarse en su forma binaria (bits). El número 65 y, consecuentemente, la letra A, se expresa en binario como 01000001. La B es 01000010. Si se envía el mensaje AB, se transforman primero los caracteres en su equivalente binario: 0100000101000010.

A continuación, Alice genera una clave, un número aleatorio que tiene la misma longitud que el mensaje. Los ordenadores y algunas calculadoras pueden generar números aleatorios. Incluso existen libros sobre números aleatorios.

Este sistema se llama *one pad* porque tradicionalmente se tenía anotado en un bloc o en un libro las claves aleatorias en cada una de sus páginas. Se usaba la primera clave para el primer mensaje y después se arrancaba la hoja; se usaba la segunda clave y se arrancaba la hoja, y, así, sucesivamente.

Con frecuencia, los ordenadores y calculadoras proporcionan números aleatorios en forma de fracciones pero sólo tiene que multiplicar la fracción por el número que le corresponda para obtener el número entero que necesite.

He usado la función de mi calculadora Casio RAN# para generar dos números aleatorios; 0.69 y 0.23, que multiplicados por 100, da un resultado de 69 y 23. Transformados en binarios y colocados uno al lado del otro, dan la siguiente clave: 0100010100010111.

A continuación, Alice combina (suma) la clave y el mensaje (no intente imaginarse ninguna suma binaria):

0100000101000010 (texto plano)

+ 0100010100010111(clave)

0101100100100001 (texto cifrado)

El texto cifrado puede enviarse ahora a Bob, con Eve escuchando.

Bob puede descifrar el mensaje porque tiene una copia de la clave y tan sólo tiene que restar la clave al texto cifrado para obtener el original.

0101100100100001 (texto cifrado)

- 0100010100010111(clave)

0100000101000010 (texto plano)

Eve ha estado escuchando todo, pero sólo tiene el texto cifrado porque fue lo único que se envió: 0101100100100001. Por supuesto, siempre puede construir un programa que, por la fuerza bruta, intente deducir todas las claves posibles (0000000000000001, 0000000000000010, 0000000000000011 y así hasta 1111111111111111). Sin embargo, así obtendrá muchos mensajes distintos. Tendrá muchos mensajes con sentido. ¿Cuál es el real? El que dice "Pon el paquete en la cabina de teléfonos" o el que dice "¿Salimos esta noche?".

Es el viejo problema de los infinitos monos. Un número suficiente de monos aporreando un teclado el tiempo suficiente no sólo podrían llegar a producir un Hamlet, sino también todas las posibles versiones de Hamlet, incluso aquella que sólo falla en la frase: "Ser o no ver" o "r o no ser". U otra perfecta excepto en la frase: "Esa es la canción". Y así cualquier variación de Hamlet, cualquier obra o cualquier escrito realizado.

Es como si recorriese todas las bibliotecas del mundo y apilara todos los pequeños trozos de papel de fumar en un montón de letras y, a continuación, multiplique ese montón un billón de veces. Ahora, entre en esa montaña de papeles del tamaño de una galaxia y comience a mirar alrededor. Encontrará allí todas las posibles combinaciones de letras, palabras y obras de Shakespeare, si es que tiene tiempo.

18.3.2. ¿Cómo se envía la clave?

El sistema de *One time pad*, perfecto como es en teoría, requiere obtener tanto de Alice como de Bob, la clave sin que se entere Eve. Ahí está la dificultad.

Ese es el principal problema de este sistema perfecto, al igual que sucede con muchos otros, como DES, por ejemplo, en los que las personas deben transmitir las claves secretamente.

Entre en la distribución cuántica. Puede resolver todos los problemas sobre la distribución de claves. Actualmente, los científicos envían claves cuánticas en unos kilómetros (sobre 60 kilómetros es el record actual), pero quizá puedan enviarse pronto por todo el mundo. Y, cuando esto se haga, probablemente estemos ante el último sistema criptográfico indescifrable.

Las técnicas actuales de transmisión de fotones no son lo suficientemente potentes para ser usadas en la transmisión de claves en un dispositivo real. Sin embargo, se están realizando progresos. Se están llevando a cabo pruebas para el envío a través de conexiones tierra-satélite-tierra. Ya veremos.

El perfecto sistema criptográfico indescifrable

Sí, es verdad. A través de la historia, casi todo aquel que ha inventado un sistema criptográfico ha afirmado que es perfecto y que jamás nadie será capaz de descifrarlo. Con una sola excepción, el sistema *one-time pad*, estas afirmaciones siempre han sido falsas. (Se llama *pad* porque en una implementación del sistema tanto el emisor como el receptor utilizan un *pad* (bloc de notas) en el que las páginas contienen cada una claves aleatorias. Cuando se utiliza una clave, a continuación, se arranca esa página (o se come, o se le dispara veinte veces o se toma alguna otra medida drástica).

Por supuesto, los conocidos sistemas DES y RSA, no han sido todavía descifrados, pero, ¿cuánto tiempo cree que tardarán en hacerlo?

Así que, ¿cuál es el poderoso motivo que me permite afirmar que el sistema que voy a explicar en este capítulo y que proporciono en el CD es, de hecho, perfecto e indescifrable?

Mi afirmación no es completamente irracional. El sistema recogido en el programa descrito al final del capítulo es una variación de *one pad*, y este tipo de sistemas es perfecto en teoría. La razón de que no tengan un amplio uso es porque son ineficientes para cifrar mensajes a gran escala (por ejemplo, en la guerra). Sin embargo, los *one time pad* han sido utilizados para cifrar comunicados en la línea Washington-Moscú y para otras transmisiones de especial relevancia.

El sistema que encontrará en este capítulo, al que llamaremos el *one time pad* aleatorio o ROP, puede ocultar perfectamente las entradas a su diario. No importa lo activo que sea; su vida diaria no generará datos suficientes como para saturar el sistema ROP.

19.1. Los problemas de *one time pad*

Los sistemas *one time pad* no son ajenos a los problemas clásicos de todos los sistemas criptográficos basados en claves. Si quiere generar una buena clave aleatoria y larga, tendrá que encontrar una manera para transmitir la clave de forma segura entre las dos personas que usen el mensaje cifrado. Alice crea una clave aleatoria, la usa para cifrar su mensaje y envía el mensaje a Bob. Sin embargo, debe crear ahora una manera de transmitirle la clave a Bob sin que la mirona de Eve la intercepte. Si Alice transmite la clave sin cifrarla, obviamente Eve también la va a poder usar. Si Alice cifra la clave, simplemente ha conseguido que el problema de la transmisión de la clave retroceda un paso. ¿Cómo enviará la segunda clave que descifra a la primera?

***Nota:** Si está interesado en técnicas que aseguren transmitir con seguridad una clave secreta a través de un medio seguro, busque en el protocolo Diffie-Hellman.*

Un sistema *one time* (una sola vez) es completamente indescifrable siempre y cuando se use una clave absolutamente aleatoria, tan larga o más que el mensaje cifrado, y diferente para cada mensaje enviado. Puede ver cómo la generación de estas claves podría ser práctica si sólo se cifran unos párrafos al día (su diario), pero imposible si tiene que enviar decenas de miles de párrafos al día (durante una guerra). Por un lado, habría que generar una clave aleatoria gigante (una longitud de miles de párrafos) y, además, estaría la pesadilla de intentar coordinar y asegurar la transmisión de las claves.

En la práctica, si trabaja sólo a nivel de tráfico diario, no hay razón alguna para preocuparse por la transmisión de la clave (sólo el dueño puede abrir el diario). Y, siendo prácticos, querrá generar una clave que pueda recordar y usar cada vez que entre en el diario, en vez de tener que generar una por entrada. Sin embargo, utilizar una única clave memorizada repetidas veces debilita el sistema y hace, teóricamente posible, que otro la descifre.

Esto es de verdad. Aquí tiene para su uso personal este código indescifrable, perfecto. Si asume el trabajo de generar claves aleatorias por cada nuevo mensaje, sus datos estarán perfectamente seguros. No habrá patrones ni nada similar

en el texto cifrado. De este modo, un ataque por la fuerza bruta lo único que puede hacer es producir todas las entradas posibles al diario que viene a ser lo mismo que no producir ninguna (¿cómo puede saber el intruso cuáles son las reales?).

En este capítulo, describiré el sistema ROP y le proporcionaré un programa informático que automáticamente cifra y descifra. Es fácil de usar (a no ser por el conocido problema de generar de buenas claves y mantenerlas seguras). El programa ROP se encuentra disponible en el CD, tanto como utilidad ejecutable (programa ejecutable denominado ROP.exe) como código fuente de Visual Basic para aquellos programadores que quieran ver cómo funciona o personalizarlo.

19.2. La solución: ser aleatorias

En 1994, mi compañero en el libro *The Visual Basic Power Toolkit*, Evengelos Petrou, y yo desarrollamos el sistema ROP y lo publicamos en ese libro. Ofrecimos un premio de 200.000 Ptas. a la primera persona que pudiese descifrar el mensaje cifrado que publicamos en el libro. Nadie lo hizo.

¿Cómo construimos el sistema de cifrado ROP? Queríamos que fuera lo más resistente posible a un intento de descifrado por parte del enemigo. Usamos Xor, al igual que muchos sistemas actuales de cifrado informático, pero no proporcionamos la clave. Si se ejecuta la orden Xor entre una clave y el texto original, ésta queda parcialmente revelada cuando concurre un patrón más largo que la clave dentro del texto plano.

19.2.1. Cuidado con Xor

Por ejemplo, después de ejecutar el comando Xor, un modelo repetitivo de números reales (los dígitos binarios 1 y 0, como en 1010101010101010) revelan, por lo menos, la longitud de la clave y la mitad de los símbolos que hay en ella (ya que Xor con cero no cambia nada). Otra forma más sutil puede tener lugar si el sistema de cifrado rechaza cifrar un carácter en sí mismo, es decir, *A* nunca podría ser cifrado como *A*. En ese caso, el intruso puede ejecutar un análisis normal de frecuencia pero buscar los símbolos menos usados.

Para evitar los problemas de la mayoría de sistemas criptográficos, usaremos un generador aleatorio de números en Visual Basic. El comando RND puede proporcionarle series de números aleatorios que, si la aleatoriedad es perfecta, no revelará ningún modelo de distribución de frecuencias en el rango 0-255 de los distintos caracteres que usamos en nuestro código (El sistema aleatorio de Visual Basic no es perfecto, pero servirá.)

19.2.2. Curiosidades sobre RND

Un uso repetido de RND genera una lista de números aleatorios. Sin embargo, cada vez que ejecuta un programa de Visual Basic con un RND dentro, la lista que genera RND es idéntica. Es verdad que, dentro de la lista, los números son aleatorios pero, si se ejecuta el programa cinco veces, se obtienen las mismas listas.

Por ejemplo, si ejecuta:

```
For I = 1 to 3
    Print Rnd(4)
Next I
```

obtendrá:

```
.7055475
.533424
.5795186
```

Ahora, detenga el programa. A continuación, ejecútelo y obtendrá la misma secuencia de números.

También es significativo que no pueda modificar la orden RND. Compruébelo dándole distintos números para ver si puede conseguir resultados diferentes.

Si se le proporciona al comando RND un argumento diferente, sigue produciéndose la misma secuencia. RND ignora cualquier valor positivo dentro del argumento. Por ejemplo:

```
For I = 1 to 3
    Print Rnd(1)
Next I
```

obtendrá:

```
.7055475
```


.533424
.5795186

El comando RND es insensible a este argumento. Podría usarse cualquier número positivo como un argumento y aún así se obtendría la misma secuencia como resultado.

19.2.3. Un bloc enorme

Piense en RND como en un libro que contiene en cada página un número aleatorio. El quinto dígito aleatorio de la página 1 siempre es el mismo y el cuadragésimo cuarto de la página 48 también. Es una larga lista invariable. Es como una guía telefónica infinita pero, en vez de nombres y direcciones, sólo hay números aleatorios.

El algoritmo generador de números aleatorios ya ha sido usado en varias versiones de Visual Basic durante más de 20 años. Esta es ya una conocida lista de números y cualquiera puede verla simplemente imprimiendo los resultados de RND:

```
Sub Form_Load ()  
Show  
For i= 1 To 10  
    form1.Print Rnd  
Next i  
End Sub
```

Nuestro objetivo será movernos de forma impredecible a través de las secuencias predecibles de números obtenidos con RND. De cada vez, me moveré hacia adelante en una distancia especificada por el código ASCII de un carácter en nuestra clave y también la distancia especificada por el código ASCII de un carácter en el texto original.

Por ejemplo, el código ASCII asigna a la letra *b* el valor 98, de forma que si hay una *b* en la clave, me moveré 98 lugares a través de la lista de los números aleatorios de RND.

De esta forma, un intruso puede conocer esta lista, pero no saber hasta qué punto he avanzado de cada vez. Nos movemos hacia adelante una distancia aleatoria por cada carácter del texto original. También nos vamos desplazando por todos los caracteres de la clave y, si la clave es más corta que el texto original, empezamos otra vez con la primera letra de la clave. En cada paso, ejecutamos la orden Xor so-

bre el número correspondiente de la lista y el carácter correspondiente del texto original.

Este enfoque permite obtener dos ventajas particulares. Primero, si alguien se las arregla para averiguar la posición de partida correcta dentro de la lista de números, pueden conseguir traducir ese primer carácter del mensaje original. Pero esto no le ayuda en localizar la siguiente posición.

Del mismo modo, si un intruso tiene la suerte de averiguar las dos primeras localizaciones correctas del libro, no habrá obtenido ninguna información útil sobre la localización de la tercera posición. Lo que se encontrará es que no hay ningún patrón que pueda seguir. Nuestro mensaje está esparcido por toda la lista.

El intruso puede suponer que el punto de partida estará centenares de posiciones más abajo (como demostraré posteriormente). Pero, aparte de eso, el efecto de esparcimiento se vuelve cada vez más complicado hasta que se igualan las probabilidades de que aparezca cualquier número a medida que descendemos en la lista.

La segunda gran ventaja de nuestro enfoque es que el resultado, el documento cifrado, no tiene patrones. Gracias a la aleatoriedad de los números (de la lista de RND) que estuvimos empleando para ejecutar la orden Xor con el texto plano, el documento cifrado carece de patrones y es estadísticamente neutro. Recuerde que, tradicionalmente, un intruso descifra un código contando la frecuencia de aparición de los símbolos del documento cifrado o usando otros métodos (como el análisis de la frecuencia columnar descrito anteriormente en la sección titulada "Construir una antitabla"). Nuestro algoritmo elude estos enfoques. Debemos dar un paso más porque estamos haciendo público el algoritmo de ROP y la lista de números aleatorios también es pública.

19.3. La historia de la máquina

Nuestro paso extra es incorporar el historial de la máquina en cada interacción de nuestro bucle principal. Encontrar una posición correcta en el libro no proporciona ninguna pista sobre otras posiciones correctas subsecuentes, aparte del hecho de que la siguiente posición está entre un intervalo determinado y otra posición más abajo en la lista.

Este intervalo, por supuesto, es finito y se conoce porque los códigos ASCII que se pueden usar están limitados a los

caracteres que se pueden escribir desde el teclado del ordenador. El intervalo es entre el carácter de espacio en blanco (32) y la tilde ~ (126). Por lo tanto, el intruso sabe ya que se debe mover por lo menos 32 posiciones a través del listado de números, pero no más de 126 posiciones. De todas formas, esto no reduce el problema de la falta de patrón. La sincronización con el modelo es totalmente imposible sin saber la clave.

19.3.1. Localizar la primera posición

Así funciona el sistema. Primero el usuario escribe una clave. Recuerde que cada vez que utilice el comando RND en un programa que se está ejecutando, puede moverse una posición hacia adelante dentro de la lista invariable de números aleatorios (uno más abajo en la lista). Nuestro sistema de cifrado utiliza el código de valor de cada letra en ASCII para crear la clave y determinar hasta qué punto debemos descender en la lista de números.

Si la primera letra de la clave es *A*, Visual Basic nos puede decir (con el comando ASC) que la *A* mayúscula tiene el valor de 65 en el código ASCII. De forma que "abrimos" la lista de números aleatorios y movemos nuestro dedo hacia abajo a través de la misma hasta que llegamos al 65º número aleatorio.

19.3.2. Un mensaje estadísticamente plano

Ahora tenemos ya nuestro punto de partida. Este es el momento en el que empezaremos a usar la lista de números para cifrar el texto. Como verá, este enfoque tiene como resultado un mensaje cifrado que no tiene ningún patrón de frecuencia. No habrá nada que analizar en el mensaje cifrado. Cada carácter entre 0 y 255 tendrá el mismo porcentaje de frecuencia.

Si se cifra un mensaje largo (por ejemplo unos 30.000 caracteres) y contamos a continuación el número de veces que aparece el valor 212, el resultado obtenido será 1/255 de 30.000 ó 118 veces. Se encontrará con que dos valores comprendidos entre el rango de 0 a 255 aparecen con la misma frecuencia. El mensaje secreto es estadísticamente plano y no revela nada de su contenido.

Simplemente, no habrá ningún patrón que pueda relacionar el mensaje con la lengua en la que fue escrito. No

habrá frecuencia de *e* o *th* (en el caso del inglés), espacios en blanco o cualquier otro texto que el intruso pueda descubrir. Tras ejecutar Xor, cualquier letra se convertirá en cualquiera de los 256 símbolos. Y debido a que la clave cambia dinámicamente, nunca se sabrá su longitud. Ni tampoco dejará al descubierto los ceros u otros posibles patrones que se repitan tras Xor.

19.4. Trabajar la clave

Aquí está la subrutina donde tomamos la clave de usuario, a partir de la cual determinamos nuestro punto de partida en el libro de números aleatorios:

```
Private Sub getkey(key)

Form1.Hide
Form3.show
DoEvents 'refresh display of Form3

Dim X As Integer
Dim z As Single
Dim t As Single
Dim n As Long

For i= 1 To Len(key)
    x= Asc(Mid(key, i, 1))
    Form3.Caption = X 'display it on Form3
    n=i*X
    For j = 1 To n
        t = Rnd
    Next j
Next i

End Sub
```

La variable *t* es un tipo de dato punto flotante con precisión simple. Es el tipo de dato más largo al que RND es sensible. Fíjese que nos movemos desde 1 hasta la longitud de la clave (la variable denominada clave), tomando como referencia el valor de cada carácter en el código ASCII. Para cambiar un poquito más las cosas, multiplicamos ese código por la posición del carácter dentro de la clave.

$$n = i * x$$

Ejecutamos el comando RND un número n de veces. No nos importan los números aleatorios que se devuelven usando repetidamente RND en este punto. Ignoramos los valores puestos en t . Todo lo que queremos hacer es usar repetidamente el comando RND para desplazarnos por la lista de números aleatorios. Cada vez que se utiliza el comando RND, nos posiciona una entrada más adelante en el libro de números, como si un dedo se fuera desplazando hacia abajo en el listín telefónico.

Una vez que tenemos nuestra posición de partida, dejamos esta subrutina y volvemos al bucle principal (explicado a continuación). Ahí procederemos al cifrado real del mensaje.

Cada vez que utiliza el comando RND, Visual Basic le proporciona un valor único distribuido aleatoriamente (es un número fraccionario). Ejecutamos la orden Xor entre ese número y cada carácter del documento original que queremos cifrar.

Recuerde que, durante los procesos de cifrado y descifrado, utilizamos el código ASCII de cada carácter de la clave para determinar cuántas veces tenemos que ejecutar RND (y, por lo tanto, cuánto nos tenemos que desplazar a través de la lista de números) para así localizar el siguiente valor y usar el comando Xor. Ahora que hemos desplazado nuestro punto de partida dentro del "listín" de números, ha llegado la hora de cifrar o descifrar el documento.

19.5. El corazón del programa

Aquí tiene el núcleo del programa de cifrado. A esta función de cifrado se le proporciona *Archivo*, el nombre del archivo que vaya a ser cifrado. (Si el archivo ya está cifrado, lo que hará el programa será descifrarlo. Este programa trabaja simétricamente en ambas direcciones, construyendo la misma lista aleatoria cada vez, basada en la clave, que también se proporciona para la función de cifrado.)

```
Private Function encrypt(file, key)
```

```
Dim FileLength As Long, v As Long
```

```
Dim lk As Integer, i As Long
```

```

lk = Len(key)

On Error Resume Next

getkey key

Dim c As String * 1

If UCase(Right(file, 3)) = "CRP" Then
    ext = ".CR2"
Else
    ext = ".CRP"
End If

p = InStr(file, ".")

If p = 0 Then
    outfile = file & ext
Else
    outfile = Mid(file, 1, p- 1) & ext
End If

FileLength = FileLen(file)

Open file For Binary As 1
If Err Then MsgBox (Error(Err))
Open outfile For Binary As 2
If Err Then MsgBox (Error(Err))

mask = Int(Rnd * 256)

'main loop

For i= 1 To FileLength

Get 1, , c
c = Chr(Asc(c) Xor mask)
Put 2, , c

alltext = alltext & c
Form3.Text1 = alltext `display

```

```

    If i Mod 100 = 0 Then
        v = (i / FileLength) * 100
        Form3.Caption = Int(v) & "%"
    End If

    'rotate key

    key = Right(key, 1) & Left(key, lk -1)

    ' get new leftmost character ASCII value

    X= Asc(Left(key, 1))

    'throw away random numbers up to the value
of the character

    For j = 1 To X
        t = Rnd
    Next j

    mask = Int(Rnd * 256)
Next i

Close 1
If Err Then MsgBox (Error(Err))
Close 2
If Err Then MsgBox (Error(Err))

Cr = Chr(13) & Chr(10)
Form3.Caption = "Successful!"
Form3.Text1 = "The result has been saved as:
" & UCase(outfile) & "." & Cr & Cr & "Many
cyptertext characters are unprintable, so the
following display of the newly created file's
contents may seem abbreviated. However, the
total number of characters processed is: " &
Len(alltext) & Cr & Cr & Cr
Form3.Text1 = Form3.Text1 & alltext
End Function

```

Esta función cifra cada carácter del archivo del texto original. Como es normal, hay algunos aspectos que hay que

tener en cuenta. Definimos algunas variables. La longitud del archivo nos dice cuántos caracteres contiene el archivo. La variable *v* se usa para visualizar el progreso del cifrado dentro del formulario en tanto por ciento. La variable *lk* corresponde a la longitud de la clave. La variable *i* es nuestro contador para el siguiente bucle.

Hallamos primero la longitud de la clave de usuario:

```
lk = Len(key)
```

A continuación, utilizamos la `getKey Sub` para movernos a la posición inicial dentro del libro de números basado en los valores acumulativos de ASCII de todas las letras de la clave (la subrutina se describió anteriormente en este capítulo en la sección "Trabajar con la clave").

```
getKey
```

A continuación, definimos la variable *c* como un carácter de un único byte (lo que nos posibilita mover datos de un archivo byte a byte).

```
Dim c As String * 1
```

En las líneas siguientes definimos un nuevo nombre de archivo que será el archivo final con la extensión .CRP. Si el archivo .CRP ya existe, no lo sobrescribiremos. Por el contrario, crearemos otro archivo de extensión .CR2. Esta es una pequeña medida de seguridad. Evitará "castigar" al usuario si introduce mal la clave durante el proceso de descifrado (lo que cifraría aún más el documento y el usuario podría pasar un mal rato intentando recordar la palabra tecleada cuando introdujo incorrectamente la clave).

A continuación, descubrimos el número de caracteres:

```
FileLength = FileLen (file)
```

El proceso se completa cuando abrimos el archivo original (*Archivo*) para leerlo y un nuevo archivo *Archivo resultante* para escribir. En el último es donde almacenamos los resultados, el documento cifrado.

A continuación, usamos el comando `RND` para generar un número aleatorio entre 0 y 255 y poner el resultado en la variable *mask*, que será usada para ejecutar `Xor` con un carácter de texto plano, produciendo así un carácter cifrado.

```
mask = Int (Rnd*256)
```

Llegados a este punto, introducimos el bucle principal. Por cada carácter del archivo fuente, conseguimos el siguiente

te carácter, ejecutamos Xor con una nueva mask y lo almacenamos en el *Archivo resultante*. A continuación, mostramos nuestro progreso en el formulario de modo que el usuario no crea que el programa ha dejado de funcionar:

```
For i= 1 To FileLength
    Get 1, , c
    c = Chr(Asc(c) Xor mask)
    Put 2, , c

alltext = alltext & c

If i Mod 100 = 0 Then
    v = (i / FileLength) * 100
    Form3.Caption = Int(v) & "%"
End If
```

Ahora, rotamos la clave, como si empujáramos un cabalito del tiovivo. La última letra de la clave pasa a ser la primera:

```
key=Right(key, 1)& Left(key,lk-1)
```

Y obtenemos el valor del código ASCII del carácter de la clave que está a la izquierda:

```
x=Asc(Left(key, 1))
```

A continuación, nos movemos en el libro de números. La distancia que nos movemos es el valor de esa letra de la clave dentro del código ASCII:

```
For j= 1 To x
    t= Rnd
Next j
```

Usando RND, creamos una nueva máscara:

```
mask=Int(Rnd*256)
```

Finalmente, se repite el bucle entero hasta que cada carácter del archivo original haya sido cifrado (o descifrado) y almacenado en el *Archivo resultante*:

```
Next i
```

El proceso de cifrado/descifrado es simétrico, gracias al carácter alternante de Xor y a la lista estable de números

aleatorios. Recuerde que el comando Xor alterna entre dos estados. Por lo tanto, cuando ejecutamos un archivo de texto sin formato a través de este programa, obtenemos el resultado de cifrado. Si ejecutamos ese resultado a través del programa una segunda vez (después de escribir la clave correcta), obtenemos de nuevo el original. El programa no sabe ni le importa si algo ha sido cifrado o descifrado.

En otras palabras, conseguiremos la misma secuencia de números del libro de números cada vez que queramos ejecutar este algoritmo de cifrado/descifrado (suponiendo que se utilice la misma clave cada vez). El texto quedará encriptado la primera vez que el programa lo procese, descifrado la segunda, cifrado la tercera y así sucesivamente.

***Nota:** Fíjese que no puede cifrar un archivo y, a continuación, con el programa ROP todavía funcionando, descifrarlo (o cifrar otro archivo). Hay que ejecutar de nuevo el programa para que el "dedo deslizante" vuelva a estar en su punto de partida en la lista de números. Si no se ejecuta de nuevo, el programa le avisa del problema y se cierra automáticamente.*

19.6. Cómo utilizar el programa ROP

Siga estos pasos para cifrar un archivo usando el programa ROP (puede ser cualquier archivo: un gráfico, un documento de Word .DOC, un archivo .TXT del Bloc de Notas o cualquier otra cosa).

***Nota:** Si olvida su contraseña, no hay forma de que pueda descifrar un archivo que previamente cifró utilizando el programa ROP. Sin embargo, el programa ROP no borra el archivo original por lo que éste se conserva. Pero, de la misma forma, si está seguro de que recuerda su contraseña, querrá borrar el archivo de texto original. De lo contrario, ¿qué objeto tendría el llevar a cabo el cifrado?*

1. Ejecute ROP.exe desde el CD que se adjunta al libro; también puede copiarlo y ejecutarlo desde su disco duro.
2. Verá la lista de unidades de disco, la lista de carpetas y la lista de archivos.

3. Utilice estos cuadros con listas para localizar en su disco duro el archivo que quiere cifrar (Haciendo clic sobre los elementos, los selecciona o los activa.)
4. Haga doble clic sobre el nombre del archivo que quiere cifrar. Aparece la ventana en la que se solicita la clave.
5. Escriba la clave (recuerde que reconoce la diferencia entre mayúsculas y minúsculas).
6. Haga clic sobre aceptar. Se cierra la ventana de la clave y ahora verá la ventana de resultado.
7. La localización de partida (en la lista de números) se ha alcanzado y, por lo tanto, se ha cifrado el texto original. A través de estos dos procesos, el título en la barra de la ventana Result muestra la actividad, para que no se preocupe de que le suceda algo al programa.
8. Cuando se haya realizado el proceso de cifrado, se muestran los resultados.

***Nota:** El proceso de descifrado es idéntico al de cifrado descrito en los pasos anteriores. La única diferencia es que hay que hacer un doble clic sobre un archivo cifrado (con la extensión .CRP o .CR2) y el resultado que se muestra en el paso 8 es el texto plano.*

Recuerde que el CD de este libro tiene los códigos de los programas ROP.exe y de Visual Basic para el programa Encryptor.

19.7. Un cifrador práctico

Aquí tiene un programa práctico para proteger de ojos curiosos su contraseña, sus secretos financieros u otras cosas. Este programa no ofrece el nivel de seguridad de RSA. Ni tampoco proporciona el cifrado perfecto disponible a través del programa ROP presentado en este capítulo. Sin embargo, si usa el programa Encryptor, nadie que tenga acceso a su ordenador podrá descifrar la información que tenga almacenada (esconda este libro o también verán los secretos descritos a continuación).

Escribí este programa para mí porque quería realmente una forma rápida de ocultar información, notas para mí mis-

mo, todas esas contraseñas que necesito para Internet y otras cosas que no me gustaría que viera nadie.

Para instalarlo, localice la carpeta de Encryptor en el CD de este libro. Copie todo el archivo en su disco duro. Ejecute el programa Setup.exe. Es todo lo que tiene que hacer.

Antes de que utilice por primera vez Encryptor, fíjese que, a menos que sea un programador de Visual Basic, tendrá que ejecutar el programa desde C:. Si utiliza un disco duro con una letra diferente, tendrá que ajustar el código fuente de Visual Basic (vea las explicaciones a continuación) y recompilar el programa Encryptor.exe.

Antes de usar Encryptor, localícelo con el Explorador de Windows, arrastre el nombre de archivo Encryptor.exe y suéltelo en el botón de Inicio de Windows. Esta operación lo incluye en el menú Inicio, haciendo posible que se le proporcione una combinación de teclas. Haga clic sobre el botón de Inicio. En el Menú Inicio, haga clic sobre el botón derecho en Encryptor.exe. Escoja Propiedades del menú que se despliega. Haga clic sobre la pestaña **Shortcut** en el cuadro de propiedades de Encryptor.exe. Haga clic sobre el cuadro de texto **Shortcut key** y teclee *E*. Aparecerá Ctrl-Alt-E. Acepte para cerrar el cuadro de diálogo.

Ahora, cada vez que teclee Ctrl-Alt-E (sin importar el programa con el que esté trabajando), Encryptor aparecerá en la parte superior de la pantalla, listo para la acción. Esto es muy útil para tomar notas secretas, pegar contraseñas o almacenar cualquier otra información privada que tenga.

Ejecute Encryptor. Se abre con un cuadro de texto vacío. A continuación, teclee las letras **qwer** dentro de ella (no le dé a Enter, escriba esas letras en la primera línea del cuadro de texto). Tan pronto como escriba esta clave secreta, sus secretos quedan descifrados y mostrados como texto normal.

Cuando ejecuta Encryptor por primera vez, le dice que no hay información en su archivo secreto. Continúe y escriba algún secretillo dentro; a continuación, haga clic sobre el botón Encrypt. Su texto se transformará en texto cifrado. Haga clic sobre END para salir del programa. El texto cifrado se almacena en su disco duro en un archivo llamado 1000.crp. No quiere que esa información se pierda a causa de un accidente.

Cada vez que ejecute Encryptor, escriba **qwer**. Escogí esta clave porque es fácil de teclear. Sin embargo, si es un programador puede cambiar sin problema el código fuente y recompilar, y utilizar la clave que le apetezca.

19.8. Sólo para programadores

Si es un programador de Visual Basic, utilice el código fuente (también disponible en el CD) para personalizar el programa Encryptor. Puede que le apetezca reforzar un poquito el sistema, usar una clave interna diferente o andar cambiando cosas de forma que si alguien se hace con este libro, no sea capaz de descifrar su archivo 1000.crp ni de desbloquear el estado de inicio de Encryptor.

Para cambiar la clave, ajuste esta línea de código:

```
If Text1<>"qwer" Then Exit Sub
```

Para cambiar la localización del texto cifrado, ajuste esta línea del código:

```
Open "C:\1000.crp" For Random As 1
```

El formulario de Visual Basic contiene un cuadro de texto y dos botones. Aquí está el código fuente completo:

```
Private Sub Form_Load()  
key = "honker"  
End Sub
```

```
Dim key As String
```

```
Private Sub Command1_Click()  
End  
End Sub
```

```
Private Sub Command2_click() 'encrypt and save
```

```
Dim l As Integer, Counter As Integer, i As Integer
```

```
mess = Text1
```

```
Text1 = Chr(13) & Chr(10) & Chr(13) & Chr(10)  
& "Encrypting..." 'blank it
```

```
Text1.Refresh  
l= Len(mess)  
Counter = 1
```

```

Open "C:\1000.crp" For Random As 1

For i= 1 To 1

c = Mid(mess, i, 1)

a= Asc(c)
a = a+ Asc(Mid(key, Counter, 1))
Put #1, i, a
tx = tx & a
Counter = Counter + 1: If Counter > Len(key)
Then Counter = 1
Caption = Counter

Next i

Close 1
Text1 = "This encryption has been stored.
..Press the End button." & Chr(13) & Chr(10) &
Chr(13) & Chr(10) & tx
Caption = ""
Command1.Caption = "End"
End Sub

Private Sub Text1_Change() 'decrypt
If Text1 <> "qwer" Then Exit Sub

Open "C:\1000.crp" For Random As 1

Counter = 1

Do Until EOF(1)
i=i+1
Get #1, i, c
If IsEmpty(c) Then
MsgBox ("There is no data in your secret
file. You can add some now. ..then press the
Encrypt button.")
Text1 = ""
Close 1
Exit Sub

```

```

End If

a= c -Asc(Mid(key, Counter, 1))
tx = tx & Chr(a)
Counter = Counter + 1: If Counter > Len(key)
Then Counter = 1
Loop

Close 1
Text1 = tx

End Sub

```

19.9. Soluciones alternativas

Si le interesan otros sistemas criptográficos, pruebe con los servicios que ofrecen los siguientes servidores de correo electrónico: pop3now (www.pop3now.com); Hushmail (www.hushmail.com) y ZipLip (www.ziplip.com).

Una famosa utilidad de cifrado que puede probar es la llamada PGP (Pretty Good Privacy; www.pgp.com). Este programa es muy famoso porque el gobierno de los Estados Unidos prohibió su exportación. Le ofrece una selección de varios sistemas, entre los que figuran los sistemas RSA, DES y Diffie-Hellman. (Diffie-Hellman utiliza una clave secreta pero ofrece una forma de que Alice y Bob la intercambien con seguridad.) PGP es gratuito y se usa para fines no comerciales. Ofrece un cifrado bueno, pero con su manual de 220 páginas y variedad de opciones, puede llegar a ser agotador. Mi única pega a PGP es, simplemente, que es difícil de usar.

Se ha escapado el gran gusano

Una semana del otoño de 1988, cambiaron los temores de los norteamericanos. Durante décadas, habían temido un repentino ataque nuclear por parte de Rusia. Sin embargo, a finales de la década de los ochenta, la Unión Soviética de Gorbachov se vino abajo y el comunismo existente hasta ese momento se evaporó.

¿Se lo habría imaginado? Nada más empezar a disminuir este temor por Rusia, empezó a sustituirlo otro. El 2 de noviembre de 1988, un virus saltó por todo Estados Unidos y estropeó ordenadores apuntando a localizaciones clave: MIT, diversas bases militares, los laboratorios Lawrence Livermore y muchos otros lugares.

Gracias a una hiperactiva respuesta de los medios de comunicación que continúa hoy en día, el ataque al complejo militar y científico pasó a ser el temor ciudadano número uno. Era un presagio de los miedos al efecto 2000. A pesar de que los medios de comunicación exageraron en su momento sobre la amenaza real que suponen hasta ahora los virus informáticos, podría haber problemas serios en el futuro.

Todos sabemos hasta que punto se ha pasado a depender de los ordenadores, desde la realización de transacciones bancarias hasta la defensa nacional. Desde 1948 hasta 1988, los norteamericanos temieron un ataque físico. Ahora, la preocupación se centra en si algún día perderán su infraestructura informática, ahora esencial.

Algunas personas construyeron refugios durante todas esas décadas de temor a la guerra nuclear. Pero, a finales de los 80, la misma gente comenzó a cavar y hacerse con "búnkeres" para el desastre anticipado del efecto 2000. Al menos esta gente, Dios los bendiga, mantuvieron activa la

industria de los alimentos deshidratados durante los buenos y malos tiempos. Y, ¿quién sabe? Puede que algún día demuestren ser los más inteligentes.

Dicho de otra forma: a los americanos ya no le preocupa que Rusia les propine un buen puñetazo en los morros, su principal temor es perder la conciencia colectiva.

Se calcula que existen unos 12.000 virus y que se inventan unos 1.000 cada año. Si está pensando que es muy difícil que un programa antivirus le proteja de este sinfín de peligros, está totalmente en lo cierto.

La ciberguerra entre virus y antivirus comenzó en el momento en que la gente corriente tuvo acceso a los ordenadores. Hasta principios de los 80, sólo las grandes empresas y gobiernos se podían permitir aquellas enormes máquinas. Sin embargo, entre 1980 y 1985, la minituriarización dio lugar al ordenador personal y millones de personas comenzaron a comprarlo.

Algunas de esas personas decidieron que la forma más divertida y creativa de jugar con esas máquinas era escribir programas que atacasen los ordenadores de otras personas.

Las redes informáticas (ordenadores conectados entre sí) también experimentaron grandes avances a principios de los 80. La gente comenzó a usar módems (teléfonos para ordenadores) para que su ordenador realizase llamadas a las BBS o a las máquinas de sus amigos. Algunos fueron un poquito más allá e intentaron llamar a los ordenadores de grandes instituciones. Descubrieron que, generalmente, era muy fácil cruzar la dura protección de la "contraseña", tan habitual en aquella época. Se utilizaban contraseñas simples, como *datos*, *información* o *Pedro*. En un experimento, un panel de expertos acertó casi la mitad de las contraseñas que usaban por aquel entonces en los Laboratorios Bell.

Un virus puede transmitirse de ordenador a ordenador de diferentes maneras: vía correo electrónico, caballo de Troya, un implante deliberado durante una reparación, entrando sin permiso en un sistema, etc. El siguiente capítulo trata con detalle este tema. De lo que debería darse cuenta en este momento es de que en la primera mitad de los ochenta, lo único que un hacker necesitaba era el número de teléfono de cualquier entidad y paciencia para pasar una hora más o menos intentando diferentes contraseñas posibles. Mientras funcionase el número de teléfono, no tendría problema porque la mayoría de éstas estaban disponibles en BBS. Si el hacker tenía una forma de acceder a la pequeña, pero pro-

metedora, red gubernamental/académica denominada Internet, entrar como hacker en otros sistemas era la mar de fácil. En especial la gente del ámbito académico está a favor del intercambio libre de información sin preocuparse de cuestiones de seguridad.

Además, el sistema bancario se estaba convirtiendo a pasos agigantados en una jugosa diana para los hacker porque cada vez más bancos comenzaban a intercambiar fondos electrónicamente. Está claro que para los bancos es mucho más rápido y económico transferir dinero a través de simples impulsos electrónicos que enviar cheques por correo tradicional o a través de guardias de seguridad. La desventaja: ya no están ahí los guardias de seguridad para proteger las diligencias blindadas de la Wells Fargo de posibles asaltos. El robo en el mundo real es, normalmente, más palpable y más fácil de detectar que el robo vía manipulación de datos electrónicos.

20.1. La broma de los 2 millones de pesetas

La única diferencia entre tener 2.000 pesetas y 2.000.000 de pesetas en su cuenta corriente es un pequeño punto magnetizado en el disco duro del ordenador del banco o un impulso eléctrico moviéndose por la línea telefónica (los fondos electrónicos son objetivo muy atractivo para los estafadores).

Esto es lo que le sucedió a una joven inocente empleada de banco en California que fue timada por el que era entonces su novio. Llamémosles Bobby y Joyce. El chico dijo que quería gastarle una broma a su colega que trabajaba en Nueva York. Sin saberlo Joyce, Bobby había abierto previamente una cuenta corriente en un banco de esta ciudad. Le dijo que hiciera una transferencia bancaria cierto día, a cierta hora, enviando 2.000.000 de pesetas a la cuenta de Nueva York. Mi amigo se va a llevar un buen susto, ja, ja.

Joyce lo hizo tal y como Bobby le había indicado. Él estaba preparado y esperando para retirar los 2.000.000 de pesetas, tan pronto como los impulsos electrónicos comenzaran a cruzar los cables telefónicos y arrojaran su bonanza en la cuenta de Nueva York. Joyce jamás ha vuelto a ver a Bobby. El banco jamás ha vuelto a ver los millones. ¿Dónde estás, Bobby?

20.2. El Gran Gusano

Quizá el primer virus famoso sea un gusano que "accidentalmente" se escapó (eso dicen algunos) el 2 de noviembre de 1988. La idea no era hacerse rico rápidamente. El objetivo establecido era intelectual: demostrar la seguridad de los sistemas informáticos.

En 1988, Internet estaba en su tierna infancia (unía en Estados Unidos a un número relativamente pequeño de universidades, centros de investigación como *Rand*, departamentos de alta tecnología como la NASA y el laboratorio de Los Álamos). A pesar de que Internet era mucho más pequeña de lo que es ahora, el país quedó consternado y los medios de comunicación asombrados de la potencia y la velocidad con la que se había extendido el virus dentro de los cerebros electrónicos del MIT, Lawrence Livermore, Berkeley y otras muchas entidades.

En unas horas, aproximadamente 3.000 ordenadores (en algunos de los puntos clave de la comunidad científica norteamericana) se vinieron abajo y sus capacidades de procesamiento y transmisión se quedaron absolutamente bloqueadas. Los expertos estiman que los daños totales ascendieron a más de 18.500 millones de pesetas.

20.2.1. El daño psíquico

El miedo, el daño psíquico, fueron bastante peores que las pérdidas económicas. Ésta era la primera vez que el hombre y la mujer de la calle empezaron a temer que muchos ordenadores se pudieran poner muy enfermos simultáneamente, muy rápido. También fue el primer grito de alarma que ilustraba lo dependiente que se había hecho la sociedad moderna de los ordenadores interconectados. Siguieron otros miedos (el más destacado el efecto 2000) pero con el Gran Gusano fue cuando la gente empezó a temer a los virus informáticos enviados por Internet, que podían, potencialmente al menos, cortar el suministro eléctrico, el de combustible y absorber el dinero de las cuentas corrientes.

Si esos genios del MIT no pueden defender sus cerebros electrónicos de Boston de un ataque, ¿cuáles son las posibilidades de un pringado cualquiera? Por supuesto, los temores eran exagerados, no realistas e inflados por las noticias de la noche (traducido: prensa sensacionalista). "¿Conse-

guirá destruir su vida el Gran Gusano que vuela por las noches? Descúbralo esta noche en las noticias de NBS".

Aunque el usuario medio teme a los virus más bien por ingenuidad, no dejan de ser una importante amenaza para mucha gente.

20.2.2. El pequeño programa que fue capaz

Enseguida apodado *el Gran Gusano*, este demostró ser "el pequeño programa que fue capaz". Como verá en el siguiente capítulo, un gusano es una subespecie de virus informático que se va introduciendo en los sistemas. Al contrario que otros virus, como Melissa, que se autoenvía constantemente a través de correo electrónico, un gusano puede intentar entrar profundamente en el cerebro de un sistema. Los gusanos procuran, a veces, permanecer en un estado latente, para no ser detectados y aparecen sólo cuando quieren para incordiar (o algo peor). Podemos pensar en los virus del tipo Melissa como en alguien que coge un resfriado y se pone a estornudar en todas direcciones, salpicando a todo el que se encuentra alrededor y contagiando la misma enfermedad, relativamente leve.

Un gusano lo podemos comparar a un herpes: serpentea dentro del cuerpo, viajando a través de una maraña de nervios para llegar a la base de la espina dorsal para esconderse (latente la mayoría del tiempo), pero haciendo apariciones esporádicas en forma de heridas o, si no se toman las debidas precauciones, causando posiblemente graves problemas durante el parto.

El significado del término gusano ha evolucionado gradualmente. Ahora incluye virus que se reproducen rápidamente y, por lo tanto, consumen una gran cantidad de tiempo de procesamiento del ordenador, consiguiendo así que éste se ponga a sus pies. Este efecto es, en realidad, una consecuencia no buscada de errores en la programación de los primeros gusanos, comenzando por el propio Gran Gusano.

Unas palabras sobre terminología hacker

Recuerde que la jerga hacker esta en constante movimiento. Esta jerga no es en absoluto coherente, si bien suelen ser bastante creativos.

Alguna gente insiste en que la característica primaria de un "verdadero" virus informático es que invade otro progra-

ma como un parásito escondiéndose dentro del host (lo que eliminaría a Melissa como verdadero virus).

Mi consejo es que no se preocupe demasiado por estas diferenciaciones. No estamos hablando de una ciencia en el sentido estricto de la palabra, como la biología. El proceso de "hacking" y de creación de virus envuelve una combinación de arte y ciencia. No espere que el lenguaje de un hacker sea más fiable o predecible que su comportamiento.

La ciencia realmente estricta exige un alto nivel de precisión descriptiva, pero incluso entre científicos, existen grados de precisión. Seguramente habrá oído hablar de la conversación de tres científicos durante un viaje en tren en Escocia. Vieron una oveja negra pastando en un ribazo. El biólogo dijo: "Observamos que las ovejas en esta zona de Escocia son negras". El matemático dijo: "Sólo sabemos que esa oveja es negra". El físico dijo: "Sólo podemos afirmar que está oveja es negra por un lado".

Nota: *La primera gran infestación de virus fue bautizada como el Gran Gusano siguiendo los personajes de los libros de Tolkien: los dos dragones llamados Scatha y Glaurung. Estos dragones tenían la habilidad de acabar con barrios enteros y, juntos, se les llamaba los Grandes Gusanos.*

20.2.3. Los gusanos buenos

Con anterioridad al Gran Gusano, se habían empleado durante años gusanos útiles para chequeos de sistemas y otros propósitos constructivos. A un gusano benigno se le llama, a veces, agente. Viaja a través de la red, buscando recursos infrautilizados, trabajos de impresora en la cola, datos específicos, atascos en la red o dónde conseguir el mejor precio por un reproductor DVD. Lo que distingue un agente de un gusano es que, aunque los agentes viajen por la red, no se alojan en sistemas individuales ni se reproducen.

20.3. Cómo ocurrió todo

El Gran Gusano es también conocido como el ataque hacker RTM, que responde a las iniciales de su creador Robert

Tappan Morris. Irónicamente, el padre de Morris había sido un fundador de la ciencia emergente de la seguridad informática.

Tras dejar Harvard con su título de licenciado, el joven Morris fue a Cornell donde, en el fatídico otoño de 1988, estaba en su primer curso de doctorado en el programa de informática. Realizaba un proyecto de investigación en una materia que le encantaba. Naturalmente, tenía acceso a los ordenadores de la universidad y, por lo tanto, acceso a Internet.

Durante algún tiempo en octubre de 1988, Robert Morris comenzó a construir un pequeño programa que planeaba utilizar para demostrar la inseguridad de las redes informáticas. Se suponía que el programa iba a ser puesto en Internet y, a continuación, "extenderse ampliamente sin llamar la atención" (según la transcripción de la declaración de la defensa en el caso Estados Unidos contra Robert Tappan Morris).

20.3.1. Los insectos enloquecen

Cuando terminó su trabajo, lanzó el gusano en Internet desde un ordenador del MIT, en un intento de esconder su identidad. Ya sabe lo que sucedió a continuación. Habrá visto decenas de películas en las que el brillante científico activa su nuevo aparato "atómico" y, de repente, para su desgracia, todos los insectos del Valle Alamogordo se vuelven gigantes como aviones y empiezan a moverse en busca de comida. Lo peor es que casi siempre prefieren alimentarse de humanos.

No es que Morris no tomase precauciones contra un desastre similar. Quería que su gusano evitase llamar la atención. Se suponía que iba a utilizar parte del tiempo de procesamiento de un ordenador, no a bloquearlo. Se suponía que nadie se iba a dar cuenta de su existencia; de lo contrario, otros programadores intentarían "matarlo". La intención no era que el Gran Gusano se reprodujese fuertemente, congestionase y luego acabase con los sistemas en los que se introducía. Más bien todo lo contrario.

Para prever una reproducción desmesurada, Morris hizo que el gusano se detuviese y chequeara si una copia estaba ya dentro de la máquina. Si no había ningún gusano en el ordenador, se duplicaba y se metía dentro de él. Si la copia ya existía, no se reproducía.

20.3.2. Contrainoculación

Sin embargo, Morris estaba también preocupado porque los programadores de seguridad podían explotar esta característica para aparentar que el gusano estaba ya residente. Esto protegería sus programas de la penetración del "Gran Gusano". Esto no podía permitirse. ¿Qué podía hacer?

La solución de Morris fue añadir un contador al gusano. No se reproduciría las primeras seis veces que encontrase una copia existente de sí mismo en un ordenador dado. Sin embargo, a la séptima, se reproduciría e invadiría. Esta característica antiinmunidad era el punto terrible del gusano.

Morris infravaloró lo intensamente reproductivo que podía ser el Gran Gusano al liberarse en la red.

En 1988, un ordenador normal conectado a Internet se desconectaba cada diez días. Por su diseño, el Gran Gusano permanecía en la memoria volátil (no se guardaba en el disco duro), de forma que cuando se apagaba el ordenador, moría cualquier copia del gusano en ese aparato por falta de electricidad.

Sin embargo, debido a que las máquinas conectadas a Internet no se desconectaban todas al mismo tiempo, el propio Gran Gusano no moría aunque si lo hacían algunas copias aquí y allá. Posteriormente, podía reinsertarse después de que se encendiese otra vez el ordenador.

Este era el equilibrio que pretendía Morris: un promedio de vida de diez días para las copias y una reinserción cada siete veces donde la copia ya vivía. Con este método, ningún ordenador particular debería infectarse con esas copias que, comenzar a ralentizarse y, por lo tanto, llamar así la atención sobre la infección. Esta era la teoría.

20.3.3. Lo que salió mal

El cálculo erróneo en el diseño de Morris era que ese gusano se reproducía bastante más rápido de lo que él había pensado. El número de réplicas a la séptima vez comenzó a dispararse tan rápido que ni desconectando los ordenadores se pudo contener la avalancha. Pocas horas después de que se soltara el gusano, miles de máquinas empezaban a dejar de funcionar; sus cerebros electrónicos se quedaban bloqueados llenos de gusanos pululando por ahí. Según la descripción utilizada por los Fiscales del Estado "Finalmente, los ordenadores ubicados por todo el país se estropearon o quedaron catatónicos".

El Gran Gusano se detectó poco después de ser liberado (por decirlo de una manera suave). Los ordenadores infestados no empezaron a sobrecalentarse ni mucho menos, arder o algo así. Hay que recordar que hay muchos cerebritos en sitios como Livermore y el MIT, conectados día y noche, simplemente porque les gustan los ordenadores al igual que a Morris. Enseguida se percataron de que había una caída del sistema. Su orgullo les motivaría rápidamente a localizar y frenar cualquier ataque que se estuviese ejecutando sobre sus preciosas máquinas.

20.3.4. Otro intento fracasado

Una reacción similar, quizás más enérgica, tuvo lugar en los centros de pruebas nucleares, la NASA, bases militares y otras agencias gubernamentales que fueron infectadas. Todos hemos visto la película *Juegos de guerra* y nadie quiere que un hacker se acerque al botón.

Siempre hay un gran momento en esas películas de los años 50 en las que el inventor de la máquina de rayos X mira por la ventana y, para su horror, ve cómo las moscas dragón, grandes como bombas, dirigiéndose hacia la ciudad. ¡Todo ha sido culpa suya! ¡No lo sabía!

Morris se dio cuenta enseguida de que su gusano estaba fuera de control, reproduciéndose a un ritmo vertiginoso y contaminando así todos los ordenadores que encontraba a su paso. Llamó a un amigo que estaba en Harvard para preguntar qué podía hacer. Inmediatamente enviaron un correo electrónico masivo dando instrucciones de cómo matar el gusano e inocular el ordenador contra una posible reinfección.

Pero ya en ese momento había sido tan efectivo en su trabajo de interferir Internet, que el correo de Morris no fue capaz de comunicarse con nadie. Otro intento fracasado

Cinco días después de que se soltara el gusano, las cosas comenzaron a volver a la normalidad y la mayoría de los ordenadores infectados pudieron conectarse otra vez a Internet el día 6 de noviembre. Los grandes bloqueos de correo electrónico se terminaron el día 12 de noviembre y mucha gente pudo leer en ese momento el mensaje de ayuda de Morris.

Las grabaciones del Tribunal de Apelaciones de Estados Unidos explican secamente lo que le sucedió a Morris: "Morris ha sido encontrado culpable, según el veredicto del

jurado, de haber violado el artículo 18 de la Constitución de los Estados Unidos, USC Sección 1030(a)(5)(A). Ha sido condenado a tres años de libertad condicional, 400 horas de servicio a la comunidad y a una multa de 2.000.000 de pesetas más los costes de este juicio".

No parece una pena demasiado dura para una persona que, intencionadamente o no, cambió la fobia colectiva americana de la bomba atómica al colapso de Internet en unos cuantos días de noviembre de 1988.

Las variedades de virus: bombas lógicas, gusanos y caballos de Troya

Un virus es un programa informático, generalmente pequeño, que se inserta en otro programa legal, como su procesador de textos. El comportamiento de los virus informáticos es similar a la forma en la que el virus de la gripe entra en el cuerpo humano y comienza a reproducirse. Si consigue incubarse, puede causarle mucho daño y mandarlo a la cama, donde tendrá que permanecer inmóvil y hecho polvo.

Un virus informático puede llegar a tener tan sólo 90 bytes de código. Es un tamaño inferior a este párrafo (cada carácter alfabético requiere un byte de memoria; sólo este párrafo tiene unos 200 bytes y 271 caracteres, incluyendo espacios). El virus medio, sin embargo, suele andar cerca de los 2.000 caracteres.

El hecho de que un virus se inserte en otros programas legítimos implica que, a veces, estos aumentan de tamaño. Los programas legítimos no se automodifican, no aumentan su propio tamaño. De forma que una de las formas primarias en las que el software antivirus trabaja es observando qué programas cambian de repente de tamaño. Los creadores de virus suelen ser muy inteligentes. Han encontrado formas de insertar virus sin que aumente el tamaño del programa receptor, como veremos más adelante.

La mayoría de los virus tienen tres componentes primarios:

1. Una forma de auto reproducirse (de forma que se pueda extender a otros programas y a otros ordenadores).
2. El resorte para que algo extraño pase (por ejemplo, en una fecha concreta).
3. El resultado consecuencia de ese comportamiento extraño (una serie de comportamientos raros, que van

desde el despliegue de un mensaje tonto estilo "Libera a Willy" hasta eliminar la información de su disco duro).

Los virus están generalmente programados para que no comiencen a actuar de forma inmediata. Un virus necesita tiempo para reproducirse y extenderse.

Los virus pueden dividirse en dos tipos primarios según lo que infecten. El tipo más común infecta archivos, generalmente aquellos que terminan en .EXE, .SYS o .COM. Todo, desde su programa de correo electrónico hasta el Bloc de Notas, puede infectarse.

El segundo tipo más común de virus sobrepasa el nivel de programas y tiene como objetivo un elemento más importante del ordenador: el sector de arranque. Cada disco, incluidos los discos duros, tiene un sector de arranque que contiene el código de arranque (lo que significa que los virus pueden alojarse ahí también). Esto significa que un disco vacío puede albergar en su sector de arranque un virus (no tiene que haber archivos guardados en el disco para que el virus sea peligroso).

Un virus escondido en un programa comienza a funcionar (se ejecuta) cuando el programa se activa. Si tiene un virus en su copia de Microsoft Word, se ejecutará el virus cada vez que arranque este procesador. Un virus que se oculta en el sector de arranque se ejecuta cuando se enciende el ordenador. Si un disquete se inserta en el disco A: cuando se enciende el ordenador, se ejecuta el código que está en el sector de arranque (incluido cualquier código de virus ahí escondido). Si no hay disquete, la mayoría de los ordenadores se van al disco duro y ejecutan desde allí la rutina de arranque en C:, y si hay un virus, este también se ejecuta.

Cuando el virus arranca, puede moverse entre los chips de memoria e intentar infectar otros programas. Por ejemplo, puede aumentar un contador cada vez que se ejecuta (por ejemplo, el tipo de virus que cuenta hasta 100 ejecuciones antes de entrar en acción). O puede llenar la memoria RAM con montones de copias de sí mismo y, consecuentemente, reducir la velocidad de ejecución del ordenador (este tipo de virus es el llamado gusano). Se calcula que hay unos 20.000 virus moviéndose por ahí adelante y que tienen distintos patrones de comportamiento y objetivos. Algunos incluso funcionan como antivírus.

Pero recuerde que los virus son programas informáticos, aunque normalmente más pequeños que un programa nor-

mal. Los virus utilizan instrucciones de lenguajes informáticos y almacenan datos, al igual que cualquier programa normal. No tiene que preocuparse por si son superprogramas escritos por programadores mágicos. Algunos creadores de virus son buenos programadores (aunque normalmente inmaduros). Pocos de ellos son auténticos ases y, de hecho, no es difícil escribir un virus. Tampoco es difícil evitarlos o erradicarlos si sigue los consejos de este libro.

A donde se va la información

La información utilizada por el ordenador puede dividirse en dos grupos: datos y programas (denominados también códigos ejecutables). Los datos en sí no pueden hacer nada. Son pura información, como su fecha de nacimiento, la montaña más alta del mundo, el segundo apellido de su abuela, etc. El programa, sin embargo, está diseñado para hacer algo: sumar números, revisar la ortografía, calcular el impuesto sobre las ventas y muchas otras cosas.

La distinción entre datos y programas es similar a las partes de una receta. Por un lado, tenemos la lista de ingredientes (datos) seguido de una lista de instrucciones (programa) que convierte esos datos crudos en un succulento pastel.

Un ordenador puede almacenar tanto los datos crudos como los programas en discos duros, disquetes, discos zip, CD regrabables y otros soportes. La información se almacena de esta forma para poder usarla posteriormente. Cuando apaga el ordenador, se pierde cualquier información que resida en los chips de su memoria (esa es la razón por la que debe guardarse en el disco duro u otro soporte de almacenamiento permanente).

21.1. Cómo se expanden los virus

Hay varias formas de extender virus. Un modelo clásico es este:

1. Inserta un disquete que contiene un virus en el sector de arranque. Deja el disquete en la disquetera.
2. Al día siguiente enciende el ordenador, con el disquete todavía dentro, de forma que se ejecuta el virus residente en el sector de arranque.

3. El virus deposita una copia de sí mismo en el disco duro. De esta forma, puede ejecutarse sin la necesidad de que se inserte un disquete.
4. Cada vez que inserte un disquete, una copia del virus se almacena en su sector de arranque.
5. Finalmente, le presta o da el disquete con el virus a un amigo o compañero de trabajo.
6. El proceso se repite constantemente, extendiendo el virus de máquina a máquina.

Con la llegada de Internet, la situación dio un serio vuelco. Se hizo mucho más fácil extender rápidamente cierto tipo de virus. Hasta este momento, la gente accedía a sus datos desde su propia máquina. Sin embargo, ahora los ordenadores han pasado a ser "externos", recibiendo auténticas mareas de datos desde cualquier ordenador del mundo. Es similar a la extensión de la gripe desde una aldea de China al resto del mundo en cuestión de días desde la existencia del avión, cuando antes se necesitaban meses, años o décadas para que esto sucediera. Por ejemplo, algunos expertos creen que el VIH pasó de monos a humanos en zonas del África rural en la década de los 30. A finales de los 80 se expandió por todo el mundo.

Ahora le puede entrar un virus informático a través de simples documentos. En el pasado, la lectura de un documento .DOC de Word o de un mensaje de correo electrónico no transmitía virus. Era imposible que una infección llegase a través de un mensaje de texto, de la misma forma que sería imposible coger el SIDA leyendo un artículo de la enfermedad en una revista.

Desgraciadamente, ahora puede entrar un virus simplemente desplegando un mensaje en el ordenador. Más adelante en este libro se analiza detalladamente esta nueva amenaza alarmante. Ahora los datos se ejecutan. Ya no sirve la vieja distinción entre datos y programas.

21.2. Huevos de pascua y bombas

Una bomba es un tipo de virus. Hay bombas lógicas, bombas de tiempo, bombas temporales e, incluso, bombas *fork*. Es probable que los creativos autores de los virus lleguen a hacer más tipos de bombas en el futuro.

Una bomba lógica es el clásico virus que se queda por ahí inerte (en un archivo o sistema operativo) hasta que una

actuación específica lo dispara. Normalmente, una bomba lógica espera hasta que se introduzca cierta contraseña o un grupo específico de órdenes; hasta que se acceda a un archivo un determinado número de veces o hasta que tiene lugar cualquier otro proceso. Una bomba temporal es similar, pero se activa transcurrido un intervalo de tiempo determinado o en una fecha concreta).

Los programadores, incluso los de Microsoft, introducen con frecuencia códigos secretos en sus productos. La distinción entre este comportamiento y la creación de virus o bombas es mínima. Algunos programadores que escriben sistemas operativos, como Windows, casi siempre incluyen *stealth programming* (acciones que permanecen ocultas incluso a sus jefes). Normalmente es un código inofensivo, el equivalente en informática a escribir su nombre con un palo en cemento húmedo. Con frecuencia, este código es una lista de personas que han colaborado en el proyecto. Sin embargo, al final, el programador quiere que se reconozca su inteligencia. Word esconde una sorpresa y los artículos de revistas dicen cómo activar su código secreto. Los códigos personales benignos como este son conocidos con el nombre "huevo de Pascua".

21.2.1. Huevos inofensivos

Pruebe este clásico huevo de Pascua incluido en Windows 98. Haga clic en el botón derecho sobre el escritorio, elija **Propiedades** del menú desplegable, haga clic sobre **Protector de Pantalla**. A continuación, elija la opción Texto 3D de la lista disponible. Haga clic en el botón de **Configuración** (el que está al lado de la lista) y escriba la palabra *volcano* en el cuadro de texto que se despliega. Acepte, haga clic sobre **Vista Previa** y podrá ver una lista de volcanes.

O échele un vistazo a la tetera del Sombrerero Loco en el protector de pantalla de Windows 98 Tuberías. En la pestaña del protector de pantalla despliegue la lista y escoja Tuberías 3D. Acepte y haga clic sobre **Vista Previa**. Tenga paciencia. Transcurridos un par de minutos, debería ver cómo una de las juntas se transforma en una tetera, como se muestra en la figura 21.1.

Los huevos de Pascua son bastante inofensivos. Pero recuerde que muchos virus tan sólo muestran "Liberad a Willy" en la pantalla u otro mensaje similar. ¿Cuál es la diferencia entre un huevo de Pascua y un virus tradicional? La princi-

pal diferencia es que los autores originales incluyeron el huevo dentro del código, mientras que un virus invade otro código y es escrito por alguien ajeno a este. Otra diferencia es que, para ejecutar un huevo de Pascua, hay que hacer algo, mientras que la mayoría de los virus son automáticos, se activan sin que sea necesaria la intervención del usuario. Esto nos lleva a la puerta trasera.

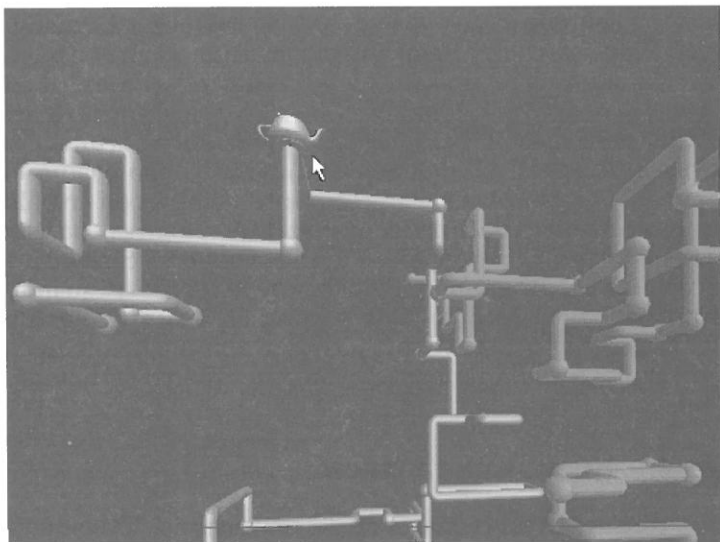


Figura 21.1. ¡Uy! Esta tetera no es algo que los jefes de programación de Microsoft esperen ver en Windows 98

21.2.2. Crucemos la puerta trasera

No todas las irrupciones en el sistema de seguridad proceden del exterior. A veces, los diseñadores de sistemas operativos o el personal de mantenimiento crean intencionadamente un agujero en el sistema. Este agujero, o puerta trasera, les permite controlar el sistema posteriormente. Por ejemplo, pueden escribir un comando y obtener una cuenta de superusuario que les da un total acceso a los archivos de todo el mundo, información sobre el funcionamiento del sistema o que les da la posibilidad de cambiar lo que les plazca. Las puertas trasera son conocidas también como *wormholes* o *trapdoors*.

Por supuesto, muchas veces es necesario que los profesionales IT u otros adquieran este nivel de control. Su trabajo es supervisar el sistema, y suelen tener motivos de peso para poder moverse por toda la red y manipularla. El problema es que, muchas veces, los extraños pueden descubrir la existencia de los pasos que abren la puerta. Pueden causar estragos.

Un segundo tipo de puerta trasera es el creado deliberadamente por un empleado enfadado. Hubo una gran preocupación en 1999 de que quizás se estuvieran insertando puertas traseras mientras los programadores preparaban los ordenadores para el efecto 2000. No se ha sabido nada al respecto pero eso no significa que no se hubieran instalado ni puertas ni agujeros. Quizá no hayan sido usados todavía, o quizás las grandes empresas ya han sufrido daños pero no quieren hacerlo del dominio público.

21.2.3. Cómo cargarse una base de datos

Se ha sabido que algunos programadores contratados o consultores han insertado bombas lógicas como una forma de asegurarse de que cobrarían por su trabajo. Funciona más o menos así: un programador está construyendo durante tres meses su sistema de facturación y conectándolo con su base de datos de inventario. Ud. se enfada porque el programador había dicho que acabaría el trabajo en dos meses. Calcula que el retraso le va a salir en unas 900.000 pesetas, de forma que las deduce del total recogido en el cheque de pago. El programador recoge el cheque, pero le lanza una extraña mirada de asesino, gris y de acero, como la de un tiburón.

Las cosas van bien durante un par de semanas y nadie se percata de la llamada de teléfono realizada al servidor de la empresa el miércoles a las tres de la madrugada. Se envían instrucciones al servidor vía telefónica y se activa la bomba lógica. Al día siguiente, toda su base de datos de inventario y de facturación desaparece por completo. Aún peor, miles de mensajes de correo electrónico han sido enviados a su lista de clientes, diciéndole en un lenguaje bastante obsceno que son de lo peor, el más @@@@&(@@@@!!!!mente pesado cliente que esta compañía ha tenido la mala suerte de topar. Creo que debería haberle pagado a aquel programador.

Una bomba temporal se activa en una fecha determinada. Ni siquiera necesita una llamada telefónica o cualquier

otra actuación para activarla. Después de poner una y haberse quedado sin cambio suficiente por culpa del cheque que le dio, el programador lo llama y le dice: "¿Me recuerda? O bien deposita el dinero que me debe en mi cuenta bancaria o el próximo lunes a las 11:22 saltará por los aires toda su base de datos. Quedará hecha trizas. Tiene una pequeña y asquerosa bomba en su sistema, colega".

Una bomba *fork* está bastante pasada de moda y nunca fue una gran amenaza. Sólo funciona en el sistema operativo UNIX. Escriba una simple línea de código y puede instalar la bomba *fork*, que crea continuamente copias de sí misma hasta que ralentiza el sistema casi hasta la parálisis. Aparte de ser un poco molesta, no provoca daños importantes. Además, al contrario de lo que sucede con las bombas lógicas o temporales, la bomba *fork* es fácilmente detectable y se puede eliminar sin ningún problema.

21.3. Caballos de Troya: nunca confíe en un griego que lleva regalos

Algún inteligente creador de virus pensó que sería una gran idea incorporar los virus dentro de algún buen software de utilidades. Mucha gente verá el paquetito brillante y, alegremente, ejecutará el programa, permitiendo así que esta desagradable sorpresa se instale.

Al contrario que muchos virus, la mayoría de los caballos de Troya no esperan dentro del sistema el momento para ejecutarse. Vayan lo que vayan a hacer, lo harán en cuanto ejecute el software que los contiene. Sin embargo, no son frecuentes, y la gente evita cada vez más utilizar software no comercial, sin importar lo atractivo que este pueda resultar. Pero fíjese que los caballos de Troya se nombran algunas veces para llamar su atención, para hacerle creer que puede conseguir algo gratis como Photoshop4free.zip o 54KPasswords.exe.

Otra subcategoría de los caballos de Troya no causa un daño en el sistema directamente visible, sino que actúa como espía, enviando al hacker datos importantes, como la contraseña. Más adelante podrá ver otros detalles de cómo cazar contraseñas.

21.3.1. Gusanos

Un gusano no necesita un programa receptor, mientras que los virus sí. Un gusano no se acopla a nada. Es autosuficiente y se autocontiene, aunque se han dado casos de gusanos que han llevado y transmitido virus.

Además, los gusanos están generalmente diseñados para viajar a través de redes, incluida esa mastodóntica conocida como Internet. En algunas ocasiones, los gusanos se plantan en la entrada de los sistemas, intentando conseguir una contraseña o localizar algún punto flaco en las defensas del sistema.

Sin embargo, la ambición primera de un gusano es reproducirse. El gusano tradicional crea una copia de sí mismo, lo que hace dos gusanos. Cada uno de ellos crea a su vez una nueva copia de sí mismo y ya son cuatro. Después 8, 16, 32... y todas estas duplicaciones tienen lugar en un abrir y cerrar de ojos. Muy pronto, el ordenador está lleno de gusanos arrastrándose por él y el sistema se ralentiza o se detiene ante el problema de sostener toda esta cantidad de vida artificial. El ordenador se va a dormir exhausto.

Fíjese que un virus típico puede tener realmente las características de las tres amenazas: el caballo de Troya, el gusano y la bomba lógica. Diversos elementos de cada uno se combinan en un paquete desagradable que contienen muchos virus: un virus se esconde en un programa aparentemente inofensivo o en el correo electrónico (como un caballo de Troya); un virus normalmente intenta reproducirse (como un gusano) y, bajo ciertas condiciones, el virus deposita su carga (como una bomba lógica). Fíjese también que algunos virus, gusanos, caballos de Troya y bombas se combinan de muchas maneras: por ejemplo, puede haber un gusano escondido en un caballo de Troya, o un programador puede haber incluido un virus dentro de una bomba. Finalmente, un virus u otra amenaza de un hacker puede tener más de un objetivo: puede, por ejemplo, combinar un sistema de ralentización del sistema con la eliminación de archivos y robo de contraseña. Al igual que sucede con la terminología hacker, las subcategorías que describen los virus son variables.

21.3.2. Huevos inofensivos, segunda parte

Recuerde que los huevos de Pascua son, por definición, inofensivos. Por el momento, ningún programador de una

empresa de software con reputación ha insertado un código maligno, bombas o puertas traseras en programas ampliamente utilizados como Windows. Por lo que sabemos hasta ahora.

Es perfectamente posible que, a pesar de los rigurosos sistemas de seguridad de los principales fabricantes de software, alguien haya insertado una bomba o un dispositivo de espionaje dentro de sus productos. La bomba aún no se ha disparado, el espionaje no ha comenzado (o sí, pero no ha sido detectado). Sin embargo, si se está recogiendo información y enviándose a través de Internet, alguien lo detectará más temprano que tarde. Más adelante se explicará cómo un cortafuegos le puede alertar cuando algún dispositivo de su disco duro intenta conectarse a Internet sin decirle que lo está haciendo (incluso programas legales "llaman a casa" con bastante frecuencia sin pedir permiso).

Cómo Melissa cambió las reglas

A la gente le han preocupado los virus informáticos desde que los ordenadores personales se popularizaron a principios de los 80. ¿Qué puede pasar con mi máquina? ¿Perderé todo mi trabajo, todos mis programas? ¿Inutilizará permanentemente el hardware de mi ordenador?

Los virus no son ni tan mágicos ni tan malos como teme la gente. Si guarda con regularidad sus documentos, lo peor que le puede pasar es que tenga que pasar algunas horas reinstalando el sistema operativo y las aplicaciones o pagarle a alguien para que lo haga.

Por el contrario, si no sabe cómo reinstalar el sistema operativo (generalmente Microsoft Windows), sus problemas pueden ser más serios. No es siempre un trabajo sencillo. Según algunos informes, todas las versiones de Windows (excepto la edición Server de Windows 2000) producidas a partir del 1 de abril de 2000 por fabricantes que tienen acuerdos de licencia con Microsoft no incluirán ya el tradicional CD con la copia de seguridad de Windows. En vez de eso proporcionan un "CD de recuperación" o "imagen del disco" de recuperación almacenado en su disco duro. Ambos enfoques para la recuperación de Windows pueden ser problemáticos.

Si tiene que reinstalar Windows, probablemente la mejor idea sea pagar a una tienda informática o a un consultor 5.000 o 6.000 pesetas para que lo hagan por usted (probablemente les lleve una hora finalizar el trabajo). En cualquier caso, su hardware no quedará permanentemente dañado por un ataque de un virus ni el virus será capaz de infectar los CD donde residen las aplicaciones originales.

A pesar del impacto que sus esfuerzos tienen en el usuario normal, los autores de virus cambian continuamente sus técnicas y cambian las reglas. Por ejemplo, el virus Melissa no es particularmente original en su carga real o en su constitución. Hubo ya otros similares.

Lo que sí es original y lo que permitió que el virus Melissa se extendiese rápidamente es su método de transferencia. Localiza su libro de direcciones de correo electrónico y se autoenvía a su lista de amigos y conocidos. Esta técnica inteligente y malvada permite al virus enmascararse como un mensaje de una fuente fidedigna de forma que pocos de sus amigos le tendrán miedo al mensaje. Como resultado, Melissa se extendió más rápido que ningún otro virus en la historia. Y, tan pronto como alguno de sus amigos abrió el mensaje envenenado, se autoenviaba de nuevo a los amigos de sus amigos. Esta es la razón por la que se extendió tan rápido.

Para hacerse una idea más clara de algunos de los comportamientos y riesgos de los virus, analizaremos detenidamente uno de los más famosos, el ataque de Melissa en 1999.

22.1. Cómo funciona Melissa

Todo comenzó el viernes 26 de marzo de 1999, con un supuesto envío del virus original al tablón de mensajes en el grupo de noticias Alt.Sex de alguien que usó una cuenta AOL. Proliferó rápidamente, pasando a ser el virus que más rápido se extendió (hasta que el virus ILOVEYOU le sacó el puesto en mayo de 2000).

Los servidores de correo electrónico de Lucent, Microsoft y otras compañías importantes comenzaron a ralentizarse ante la repentina carga que suponía la aparición de Melissa. De repente, cientos de miles de mensajes falsos de correo electrónico comenzaron a saturar la infraestructura del correo. Por primera vez, Microsoft tuvo que cerrar su vasto sistema de gestión de correo completamente para prevenir una mayor infección.

Melissa causó poco daño real (no borró ningún disco duro ni causó errores en el sistema de seguridad). Pero los daños colaterales y los costes de rastrear el virus y limpiarlo fueron enormes. En 24 horas, técnicos antivirus descubrieron que Melissa tenía números de identificación que llevarían hasta el autor (conocido como Identificador Único Global o GUID).

¿Son los ID la respuesta?

Es posible incorporar números únicos de identificación tanto en el hardware como en el software. Son los equivalentes electrónicos de las huellas dactilares y serían beneficiosos en diversos aspectos. Por un lado, si cada ordenador tuviese su propio ID, se reduciría mucho la piratería informática. Durante el proceso de instalación, por ejemplo, un juego nuevo podría memorizar el ID de la máquina y, a continuación, negarse a ser instalado en otro aparato. Del mismo modo, si todos los documentos (archivos) tuviesen su propio ID, sería más fácil llegar a sus autores a través de ellos. En teoría, esto podría decir quién ha sido el autor del virus. Aunque los defensores de la privacidad han señalado diversos puntos negativos admisibles concernientes a la posibilidad de estampar este número identificativo, tanto Microsoft Office 98 como Office 2000 asignan un ID a sus documentos. De hecho, los GUID se han estado usando desde 1985.

Sin embargo, actualmente la técnica empleada por Microsoft para el GUID (Global Unique Identifier, Identificador Único Mundial) es fácil de burlar. Al contrario de lo que sucede con una marca de agua u otro tipo de impresión de ID, que se introducen dentro de los propios datos, un GUID es un número separado, fácil de modificar, incluido en un documento típico. Cambiar el GUID es un trabajo de poca dificultad para alguien no muy iniciado en la informática. Un GUID intenta utilizar el único número de serie encontrado en la tarjeta de adaptador Ethernet. Una vez que se ha creado, el GUID se introduce en los archivos .DOC del procesador de textos, así como en los archivos de otras aplicaciones como la hoja de cálculo.

Para ser un verdadero sistema de identificación de virus, un GUID tendría que ser parte de los datos; es decir, no se podría extraer sin destruir el documento en el estuviera insertado. Es más, el GUID tendría que identificar una máquina o una dirección reales; si no está instalada la tarjeta de Ethernet, el GUID creará un número aleatorio para utilizarlo. Y, por sí mismo, un GUID sin una tarjeta Ethernet, no es útil a la hora de localizar un sospechoso. Las grabaciones de los Proveedores de servicios de Internet (ISP), como AOL, son bastante más prácticas a la hora de buscar el origen de una infección de virus.

La saga GUID no se ha acabado de ninguna manera. En octubre de 1999 aparecieron informes que decían que los GUID estaban siendo utilizados activamente por Real Network en el nuevo software RealJukebox para conseguir no sólo las identidades de la gente sino también sus preferencias musicales. Si esto es cierto, este tipo de recogida de datos puede ser utilizado para construir ciberperfiles de cada uno de nosotros y muchos usuarios opinan que este comportamiento requiere un serio debate sobre la cuestión de la privacidad.

El domingo 28 de marzo el FBI realizó una rueda de prensa y advirtió a los usuarios de Microsoft Outlook de que podrían estar difundiendo el virus y de que éste era un ataque de muy rápida expansión.

Al día siguiente, la mañana del 29 de marzo, las empresas de todo el mundo estaban nerviosas. Algunos insistían en que las comunicaciones se realizasen a través del teléfono y del papel, ningún correo electrónico; era ya demasiado tarde en muchos casos. Haga cuentas. Si Melissa puede autoenviarse a una media de 12 direcciones de correo electrónico en cada libro de direcciones de correo, no le lleva mucho tiempo cubrir toda la tierra (en teoría, Melissa estaba limitada a cincuenta nombres en cada grupo de direcciones, pero un usuario puede tener varios grupos).

El 1 de abril de 1999 fue detenido David L. Smith de Aberdeen, en el estado de New Jersey (Estados Unidos), como el supuesto autor del virus Melissa. AOL proporcionó al FBI toda la información necesaria para descubrir su identidad. A principios de diciembre Smith se declaró culpable.

22.1.1. Un ataque relativamente benigno

Como Melissa no destruye el disco duro ni causa otro tipo de daños irreversibles, hay quien lo infravalora. Ralentiza las cosas al congestionar el correo electrónico, pero es un virus relativamente benigno, simplemente interesado en extenderse sin infligir un daño permanente y serio.

Hasta que se expandió, Melissa causó pánico en muchos sitios. Incluso informáticos avanzados se alarmaron ante su velocidad de propagación y las implicaciones que esto tendría para futuros virus que tengan esta característica y que no sean tan benignos. Este virus se extiende por todo el mundo en menos de 24 horas.

No se olvide de que entre los principales sitios afectados estaban Lucent Technologies y Microsoft, que se vieron forzados a cerrar sus sistemas de correo electrónico mientras intentaban localizar la infección de Melissa. Es imposible poner precio a la cantidad de horas extras de los técnicos, las pérdidas de comercio electrónico, productividad perdida y otros gastos causados por el pequeño, rápido y masivo ataque del virus Melissa.

Melissa fue escrito en Visual Basic para Aplicaciones (VBA), un macro lenguaje que viene prácticamente con las principales aplicaciones de Microsoft, como Word, Excel y Access. VBA es un lenguaje incluido en las aplicaciones, diseñado para permitir a los usuarios la automatización de tareas habituales.

22.1.2. Lo que hace

Melissa se extiende a través de Microsoft Outlook (la hermana mayor y versión más robusta de Microsoft Outlook Express, el programa de gestión de correo electrónico que viene con Internet Explorer). Llega correo de un amigo, alguien que tiene su nombre y su dirección de correo electrónico en su libro de direcciones. El mensaje dice "Aquí tienes el documento que me pediste..." y se adjunta el documento. Por supuesto, hace clic sobre el documento adjunto porque confía en la fuente. El documento se abre y el proceso de infección comienza instantáneamente.

Antes de extenderse a otras personas, Melisa establece una "cabeza de playa" en su máquina. Inmediatamente, invade su archivo Normal.dot, una plantilla que forma la base de todos los documentos de Word. Desde aquí, se mete en todos los documentos de Word que vaya abriendo (tanto si ya existía como si lo crea nuevo). Melissa realiza esto utilizando auto macros.

Algunas macros de Word se ejecutan solas, sin que se requiera intervención alguna. Colectivamente, cada uno de los términos comienza con el prefijo *auto* (autonuevo, autocerrar, etc.). Las macros de esta especial colección se activan automáticamente en Word cuando tiene lugar una acción. Autoabrir, por ejemplo, se ejecuta cada vez que se abre un documento existente.

A continuación, Melissa ajusta algunos aspectos de la configuración de Word por defecto para facilitarse el trabajo y, finalmente, envía un mensaje de correo electrónico. El ele-

mento nuevo y peligroso del ataque de Melissa es que envía correo desde su libro de direcciones, como si el envío fuera suyo.

Los pasos de autoprotección que toma Melissa tras haber infectado su copia de Word 97, incluida la desactivación de la opción de macro en el menú Herramientas, la desactivación de la protección contra los virus en macros, así como la desactivación del prompt para guardar las características de la plantilla Normal (Normal.dot). Cada uno de estos movimientos distancia al usuario del virus porque estas características ahora desactivadas podían protegerlo de Melissa y sus efectos.

En el Word 2000, Melissa baja el nivel de seguridad a BAJO (quitando de esta manera la protección contra virus de las macros) y también bloquea el acceso a la configuración de seguridad de Microsoft. No puede ir a su configuración de seguridad, porque Melissa ha desactivado Herramientas, Macros, Seguridad. Este menú se muestra en la figura 22.1.

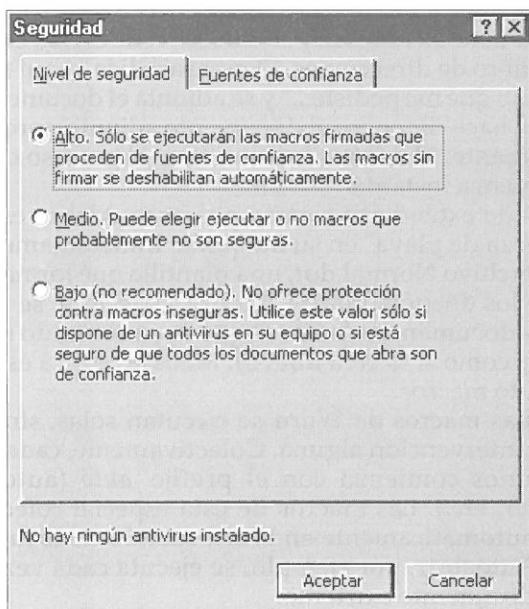


Figura 22.1. No podrá ver este cuadro de seguridad de macros si Melissa entra en su sistema

22.2. Propagación por confianza

Como verá en otras secciones de este libro, una de las formas más efectivas de hackear y de extender virus informáticos no se basa precisamente en la técnica, sino en la confianza del ser humano. La disposición de la gente para dar contraseñas, para describir medidas de seguridad y para confiar datos personales es la fuente de muchas irrupciones en el sistema de seguridad.

Melissa también se aprovecha de que, si uno recibe un mensaje de su hermano, su vecino, un viejo compañero o cualquiera que conozca, no hay nada que temer. Es un error.

Cuando se autoenvía a través de las direcciones que tiene en su libro, Melissa utiliza su dirección de correo electrónico. El mensaje aparece como suyo, porque realmente viene de su ordenador. Lo que es peor, en el campo Asunto escribe su nombre: Mensaje importante de Richard Mansfield. La figura 22.2 muestra cómo se presenta Melissa en un ordenador.

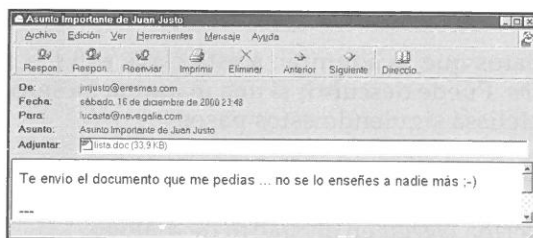


Figura 22.2. Melissa se presenta así ante un receptor inocente.

Fíjese cómo en el campo Asunto, con su nombre dentro, se recoge y se utiliza como título de la ventana que llega. El adjunto puede tener cualquier nombre aunque generalmente suele ser List.doc.

Las macros pueden hacer un montón de cosas, incluso extraer información que ni siquiera sabía que estaba almacenada. Por ejemplo, la siguiente macro ilustra cómo acceder al nombre de usuario dentro de la copia de Word.

```
Sub showauthor ()  
x = Application.UserName
```

MsgBox x

End Sub

Si quiere probar esta macro en Word, vaya a Herramientas, Macros, Editor de Visual Basic. A continuación, escriba el código anterior, comenzando en *Sub showauthor()* y terminando en *End Sub*, tal y como se muestra. Ahora haga la prueba; con el cursor parpadeante encima de la subrutina Showautor presione la tecla F5 para ejecutar esta macro. Verá un cuadro de mensaje que muestra el nombre primario del usuario de esa copia de Word.

Cuando Melissa se activa por primera vez en un sistema, empieza con el envío masivo de correo electrónico. A continuación, almacena datos en el subdirectorio HKEY_CURRENT_USER del registro de Windows (el registro es una gran base de datos mantenida por Windows que contiene información sobre su hardware, sus preferencias como la paleta de colores, el software instalado y otros detalles). Melissa chequea después ese registro y, si encuentra los datos, deja de reenviar la infección. La clave del Registro es la siguiente:

HKEY_CURRENT_USER\Software\Microsoft\Office\Melissa?

Los datos que se adjuntan a esta clave son las palabras *by Kwijibo*. Puede descubrir si una máquina ha sido infectada por Melissa siguiendo estos pasos:

1. Haga clic sobre el botón **Inicio** en la Barra de tareas de Windows y escoja, a continuación, **Ejecutar**.
2. Escriba *regedit* en el cuadro de diálogo.
3. Teclee Ctrl-F.
4. Escriba *melissa* en el cuadro de búsqueda del registro.
5. Haga clic sobre **Buscar**.

La utilidad **Buscar** analizará todo el registro para ver si ha sido atacado por Melissa.

Como incordio adicional, Melissa también chequea si los minutos (en el reloj del ordenador) coinciden con la fecha. Por ejemplo, si abre un documento a las 14:16 y es el día 16, Melissa inserta la siguiente secuencia en su documento abierto:

"Veintidós puntos, multiplicado por tres al haber hecho palabra, más cincuenta puntos por haber utilizado todas mis letras. Se ha acabado la partida. Me piro".

No sé por qué el autor (o autores) del virus escogieron este peculiar mensaje. Quizás, entre otras características

idiosincráticas, le guste jugar al Scrabble. Este mensaje raro en su documento le incordiará un poco. ¡Qué gracia! Qué perdedores.

La inserción de texto puede llevarse a cabo usando varios comandos del lenguaje de macros de VBA, incluido el comando InsertBefore.

22.3. Medidas de seguridad que ya no funcionan

Durante años, ha sido evidente que los virus se pueden extender por el correo electrónico. Con el crecimiento explosivo del comercio electrónico, de Internet y del correo electrónico, los antiguos métodos de expansión de virus, como los disquetes o los programas de shareware descargables de BBS, están obsoletos. Tal y como ilustran Melissa y el virus ILOVEYOU, las protecciones tradicionales y las viejas normas de comportamiento deben ser reescritas. Por ejemplo, aquí tiene tres medidas antivirus tradicionales que han sido recomendadas para evitar infecciones.

1. Si su programa de correo electrónico permite que los archivos adjuntos se abran automáticamente al ser recibidos, desactive esta característica.
2. No ejecute nunca software que reciba de extraños.
3. No descargue software de sitios de Internet a no ser que sepa que puede confiar en ellos. (Por ejemplo, Microsoft y otras fuentes similares prueban el software antes de ofrecerlo para descargar.)

La primera medida todavía se puede aplicar: no abra de forma automática ni documentos ni ejecutables (programa que se ejecuta, como las utilidades). La segunda necesita un pequeño anexo: no ejecute software que reciba de conocidos.

Melissa cambió las normas: ahora ni siquiera puede confiar en la información que le envían conocidos a menos que hayan acordado hacerlo previamente. Si su amigo le dice verbalmente que le va a enviar un archivo, sabe que puede confiar en él. Si, simplemente, le llega inesperadamente un archivo de un amigo, podría ser un ataque que se extiende al estilo de Melisa. Fíjese también que debe ser precavido con los archivos .DOC al igual que con los ejecutables .EXE u otros programas ejecutables. El lenguaje VBA es un poquito

más potente que otros lenguajes informáticos y las macros pueden incluirse en documentos .DOC y otros archivos tradicionales de datos. Un archivo de datos puede ejecutar y realizar diversas operaciones, incluida la eliminación de archivos y otros aniquilamientos.

22.4. El virus ILOVEYOU pasa a ser el más rápido de la historia

El virus ILOVE YOU se ha hecho con el título de ser el más rápido en extenderse de toda la historia desde Melissa (técnicamente, estos virus son gusanos, no virus clásicos).

La mañana del jueves, 4 de mayo de 2000, es el momento en que los autores del virus ILOVEYOU entraron en el proveedor de Internet Sky y dejaron salir al gusano (Sky fue capaz posteriormente de llegar al número de teléfono del apartamento de uno de los creadores). El gusano explotó. Y estas explosiones múltiples siguieron al sol por todo el mundo, infestando los servidores con copias de sí mismos y paralizando los sistemas del mundo.

El virus ILOVEYOU triunfó en su propósito porque fue una inteligente labor de ingeniería social. Aparecía como un mensaje de correo electrónico de un amigo o conocido al igual que Melissa. Pero su línea de asunto era: I LOVE YOU (te quiero). Este es un mensaje muy difícil de ignorar para la mayoría de la gente. Seamos francos, todos estaríamos intrigados si recibimos una declaración de amor de alguien que conocemos. El mensaje del virus era también muy atractivo: "Te envío una carta de amor adjunta. LOVE-LETTER-FOR-YOU.TXT.vbs".

Este mensaje es endiabladamente inteligente en muchos aspectos. Aparentemente, recibe una declaración de amor de alguien que conoce. Esto ya es motivo suficiente para tentar a la mayoría de nosotros a abrir la carta adjunta, ¿a que sí? Además, muchos se fijarían en las mayúsculas de TXT y creerán que el archivo es inofensivo. Los archivos .TXT se abren desde el Bloc de Notas y este programa no puede ejecutar un virus.

La verdadera extensión del archivo de ILOVEYOU es el .vbs que aparece al final. Sólo los programadores informáticos y un número relativamente pequeño de usuarios avanzados sabrán que esta extensión se corresponde a Visual Basic Script.

Para este grupo, .vbs es una bandera de peligro, ya que indica que el adjunto es un programa y no un documento.

El virus ILOVEYOU tiene 311 líneas de código de programa, pero ha sido el virus más rápido en extenderse y el que más daño ha causado en términos de pérdidas en productividad. Infectó a unos 10 millones de ordenadores (sobre 300 que se calcula que hay en todo el mundo). Sólo el Departamento de Estado de los Estados Unidos sumó 125.000.

Las pérdidas económicas a nivel mundial han sido calculadas entre quinientos mil millones y dos billones de pesetas. El virus ILOVEYOU le dio un mordisco pequeño pero muy profundo al comercio mundial.

Aunque las cosas pudieron haber sido peores. Como Melissa, el virus ILOVEYOU no borró discos duros. Ni siquiera revolvió en los archivos .DOC. Se dedicó a destruir tipos de archivos poco frecuentes en empresas como los archivos de música .MP3, los de gráficos .JPG y archivos con las siguientes extensiones: .js, .css, .wsh, .vbs, .vbe, .sct, .hta y .mp2.

Cuando se activa por primera vez, el virus ILOVEYOU hace algunos cambios en el Registro y busca contraseñas activas en la memoria del ordenador. Si encuentra alguna, son enviadas por correo electrónico a una página Web de Filipinas (sobra decir que ya no existe). A continuación, el virus hace copias de sí mismo y las envía a todos los que aparezcan en su libro de direcciones de Outlook Express. Finalmente, borra su página de inicio de Internet y procede a la eliminación de los archivos que tengan las extensiones indicadas antes. En su lugar, coloca una copia del propio virus.

Como suele pasar con los virus potentes, el virus ILOVEYOU tuvo unas cuantas imitaciones. No es difícil desplegar el código fuente de programación cuando llega un virus por correo electrónico. Y, cuando obtiene el código fuente, puede hacerle pequeñas modificaciones y volverlo a enviar para hacer daño a otra gente.

Una muestra del virus ILOVEYOU

Para aquellos que sepan programar, aquí va una muestra del código fuente del verdadero virus ILOVEYOU. Aunque muchos hacker afirman que el chico de Manila escribió un "simple" gusano, es realmente un programa bastante

complejo de Visual Basic. El autor (autores) entiende claramente tanto el lenguaje de programación Visual Basic como el sistema operativo de Windows. Se crean y manipulan objetos en este código, lo cual no es una técnica que pueda implementar alguien con un nivel básico o intermedio. Es programación avanzada. Pero siempre los hacker desacreditan el trabajo de otros hacker, a saber por qué. En este fragmento del virus ILOVEYOU se diseña una subrutina para modificar alguno de sus archivos.

```
Sub infectfiles(folderspec)
On Error Resume Next
Dim f, fl, fc, ext, ap, mircfname, s, bname, mp3
Set f = fso.GetFolder(folderspec)
Set fc = f.Files
For Each fl In fc
ext = fso.GetExtensionName(fl.Path)
ext = LCase(ext)
s = LCase(fl.Name)
If (ext = "vbs " ) Or (ext = " vbe" ) Then
Set ap = fso.OpenTextFile(fl.Path, 2, True)
ap.write vbscopy
ap.Close
ElseIf (ext = "js") Or (ext = "jse") Or (ext
= "css") Or (ext ="wsh") Or ( ext="sct") Or
(ext="hta") Then
Set ap = fso.OpenTextFile(fl.Path, 2, True)
ap.write vbscopy
ap.Close
```


Una de las imitaciones de este gusano pretendía ser la confirmación del cargo en cuenta del fantástico diamante que "había comprado" con motivo del Día de la Madre.

"Hemos cargado en su cuenta la cantidad de 60.000 pesetas por la compra del diamante a su madre por el día de la madre. Le adjuntamos una factura detallada en este correo electrónico".

Otra imitación pretendía avisarle, cruelmente, de la llegada de un virus peligroso.

Asunto: Aviso de virus peligroso

Mensaje: Hay un virus peligroso circulando por la red. Por favor, abra el archivo de imagen adjunto para ver cómo funciona y así poder evitarlo.

22.5. Protegiéndonos

¿Qué es lo que le protege de virus como Melissa? Su mejor apuesta es sospechar de cualquier archivo que reciba a través de Internet, sin importar quién lo envíe. Considere todos los archivos entrantes como bombas potenciales hasta que se demuestre lo contrario.

Cuando recibe un archivo adjunto a un correo electrónico, tiene las opciones de guardar el adjunto en el disco duro, abrir el adjunto o borrar el mensaje. A no ser que su curiosidad le supere, no tenga miedo y elimínelo. Simplemente rechazando abrir o guardar un archivo sospechoso es la táctica más segura y más sencilla. Ignore el archivo. Cuando elimina un mensaje de correo electrónico en Outlook Express, se guarda en la carpeta Elementos Eliminados, de forma que siempre puede recuperarlo si posteriormente el emisor le confirma que, efectivamente, ha enviado ese mensaje.

Si debe echar un vistazo al mensaje imperiosamente, guárdelo en un directorio temporal en el disco duro. Los archivos adjuntos a un mensaje siempre le dan la opción de abrir o guardar. Elija guardar. A continuación, haga que su antivirus lo analice antes de abrirlo o ejecutarlo. Y asegúrese de que su antivirus está actualizado con las últimas listas de virus. Sin embargo, recuerde que un nuevo tipo de virus, especialmente uno que se extienda rápido como Melissa o el ILOVEYOU, es probable que no sea detectado por el antivirus.

Si quiere analizar un archivo .DOC sin problema, utilice el procesador de textos WordPad de Windows. Haga clic en

Inicio, Programas, Accesorios, WordPad. WordPad ignora las macros, ya que no tiene ningún dispositivo para macros, de forma que es seguro leer archivos .DOC en este programa.

Los archivos simples de texto que se hagan para ser leídos por el Bloc de Notas de Windows terminan con la extensión .TXT. Estos archivos son seguros tanto para ser guardados como abiertos sin tener que escanearlos para ver si hay virus o sin tener que guardarlos en el disco duro. Son seguros porque la extensión .TXT hace que el Bloc de Notas abra y muestre el archivo, pero el programa no puede abrir ni macros ni scripts. Incluso si el archivo contiene un virus, el Bloc de Notas lo ignora. Recuerde que el virus ILOVEYOU engañó a mucha gente porque su archivo "tóxico" terminaba en TXT.vbs. Las tres letras finales son la extensión que Windows lee para decidir qué aplicación puede usar con ese archivo. Vbs ejecuta el Visual Basic Script y así se activa el virus ILOVEYOU y comienza a moverse por todo el sistema.

Es posible adjuntar un virus a un archivo de texto, por supuesto. Puede introducir un virus en cualquier tipo de archivo. Pero añadir un virus a un archivo .TXT sería inútil. Este nunca se activaría porque un archivo .TXT simplemente es mostrado por el Bloc de Notas, nunca ejecutado. Adjuntar un virus a un archivo .TXT no causaría más daño que estornudar contra una piedra; no importa lo grave que sea su infección porque una piedra no puede coger la gripe.

22.6. Desactivar Visual Basic Scripting

Un paso inmediato e importante que debería dar si utiliza Windows 95, 98 ó 2000 es desactivar el Visual Basic Script (también conocido como WHS, Windows Scripting Host). Esto le evitará ataques del estilo del virus ILOVEYOU. Aquí tiene cómo desactivar este programa en Windows 98 ó 2000.

1. Haga clic sobre el botón **Inicio** de la barra de tareas.
2. Vaya a **Configuración**, Panel de Control.
3. Vaya a **Agregar o quitar Programas**.
4. Haga clic sobre la pestaña de **Instalación de Windows** en el cuadro de diálogo **Agregar o quitar programas**.
5. Vaya a **Accesorios** en la lista que se muestra.
6. Vaya a **Detalles**.
7. Desactive el **Windows Scripting Host**.

8. Vaya aceptando para cerrar todos los cuadros de diálogo.

En Windows 95 tiene que hacer algo diferente para desactivar el tipo de archivos .vbs.

1. Haga doble clic en el icono **Mi PC** que tiene en el escritorio.
2. Escoja **Ver, Opciones**.
3. Vaya a la pestaña de **Tipos de Archivo**.
4. Seleccione **VB Script File** en la lista.
5. Haga clic sobre **Quitar**.
6. Si hace clic en **Sí**, lo que en realidad quiere es eliminarlo.
7. Vea si en la lista se incluye también el tipo de archivo .vbe y, si quiere, elimínelo también.
8. Acepte para cerrar los cuadros de diálogo.

22.7. Un aviso final

Algunos hacker se dedican a parlotear en los grupos de noticias de hacker acerca de los últimos ataques de gusanos. Es más, predicen un futuro escalofriante en el que se mejorará la tecnología de los gusanos: gusanos que permanecen totalmente ocultos al usuario, y que no requieren ninguna acción por parte de estos para expandirse. Peor aún, estos gusanos pueden ser polimórficos (es decir, cambian su forma constantemente para que no puedan ser detectados por ningún antivirus), independientes de plataforma (atacaran a los Mac, Linux o cualquier otro sistema operativo) y desaparecerán después de haber paralizado los sistemas de todo el mundo. Tenga miedo. Tenga mucho miedo.

Documentos que atacan (y qué puede hacer para protegerse de ellos)

Cada vez es más peligroso navegar por Internet. Como verá en este capítulo, simplemente leer un documento de texto o echar un vistazo al correo electrónico puede ser suficiente para infectar el ordenador con un virus. No sucede con frecuencia, pero puede pasar. Y es escalofriante. Imagínese tener que preocuparse por pillar una gripe simplemente por leer unos cuantos folletos de propaganda que le envían a casa.

¿Cómo puede un documento normal contener una característica que le posibilite ocultar y ejecutar un virus desde un inocente mensajito de correo electrónico de su tía Sofía? La tía es fantástica haciendo ganchillo, pero no hay forma posible, a no ser mediante trasplante cerebral, de que nuestra tía pudiese escribir un virus informático.

La culpable es una tendencia conocida como programación orientada al objeto que se ha hecho muy popular en la comunidad informática durante varios años. Entre otras cosas enfatiza el valor de poder unir datos y acciones en el mismo archivo. El texto que se lee son datos, pero ahora los diseñadores de aplicaciones informáticas están añadiendo acciones a simples archivos de datos.

Es un desarrollo bastante asombroso. Es como si en una bolsa de preparado de paella se incluyesen agua, una tartera y un hornillo. De esa forma tendría tanto los ingredientes como los instrumentos necesarios para preparar la paella.

23.1. Word nunca me haría daño

En los viejos tiempos (hace unos años), podía relajarse y dar por hecho que cuando alguien enviaba un documento informático, como un archivo .DOC o un correo electrónico, no era más peligroso que un artículo de una revista. Sólo palabras... y las palabras no pueden hacer daño.

Sin embargo, ahora Word, Excel y otros archivos de datos pueden contener un programa. Un programa es acción. Los programas son listas de instrucciones para que el ordenador haga algo.

Y los virus no dejan de ser programas diseñados para molestarle o destruir su ordenador. Incluso los documentos que aparentan ser inofensivos pueden contener virus. Ya no es necesario ejecutar un programa .EXE para infectar su máquina con un virus. Ahora, simplemente con leer un archivo .DOC en su procesador de textos, puede enviar virus a través de todo el sistema rápidamente.

Un tipo común de programa que se envía incluido en archivos de documento es la macro. Las macros son programas, generalmente pequeños, que le pueden ayudar a personalizar o automatizar ciertas acciones en sus aplicaciones, como insertar rápidamente su despedida estándar en cartas comerciales. Todas las aplicaciones más importantes de Microsoft (Excel, Word, Access) incluyen VBA (Visual Basic for Applications). VBA es un lenguaje de macros, que puede usar, por ejemplo, para que Word haga exactamente cualquier cosa que desee como usuario, incluida la eliminación de archivos y otras acciones perjudiciales de los virus.

Debido a que el uso de VBA está muy extendido y a que es un lenguaje relativamente rápido de aprender y programar, se calcula que los virus de macro son ahora el tipo más común. Mucha gente se está pasando documentos constantemente, a través de Internet o a través de la intranet de la oficina, lo que facilita la expansión de los virus de macros. Las estimaciones difieren, pero algunos expertos creen que se crean entre 15 y 20 virus de macro diariamente.

Los virus de macro más actuales están escritos en VBA, pero existen otros virus anteriores escritos en WordBasic, el lenguaje de macros de Word que ya ha sido sustituido por el VBA. Además, algunos hacker han escrito virus de macro que infectan los documentos de WordPro (el antiguo AmiPro), Lotus 1-2-3 y los documentos de WordPerfect, pero estos virus tienen mucho menos éxito que los escritos para productos Microsoft. La explicación es que WordPro y WordPerfect no almacenan sus macros dentro de los archivos de texto. En vez de eso, se agrupan juntas en un archivo aparte. Esto dificulta mucho la transmisión del virus porque la gente intercambia normalmente archivos de documentos, pero no archivos tan específicos como la colección de macros.

23.2. La mayor seguridad

Si de forma determinante no puede soportar la idea de que se infecte su sistema con virus de macro, siga los siguientes pasos para su seguridad:

1. Elija Inicio, Programas, Accesorios, WordPad.
2. Abra un documento de Word .DOC utilizando la opción de WordPad Archivo, Abrir.
3. Guarde el documento .DOC con formato .RTF, usando la opción Archivo, Guardar Como.
4. Cuando aparece el cuadro de diálogo Guardar como, escoja *Texto de formato enriquecido* (RTF) de la lista desplegable que aparece al final del cuadro, tal y como se muestra en la figura 23.1.

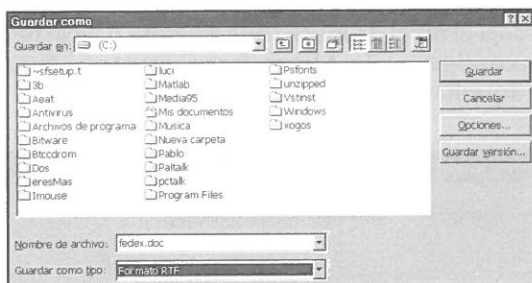


Figura 23.1. Escoja RTF como formato ya que este tipo de archivo no puede contener macros.

5. Cambie la extensión en el cuadro de texto Nombre de archivo a .RTF.
6. Haga clic en Guardar para guardar su versión .RTF del original .DOC.

Este proceso le protege de dos maneras diferentes:

1. WordPad no tiene la opción de ejecutar macros, de forma que abrir con este programa un archivo .DOC es completamente seguro, aún cuando ese archivo esté infestado de virus.
2. Un archivo .RTF no contiene macros, de forma que guardar un archivo .DOC (aunque esté lleno de virus de macro) con extensión .RTF elimina todas las macros.

23.2.1. Protección interna incorporada

Microsoft es consciente de que sus lenguajes de macro son potentes y que le están proporcionando a los hacker las herramientas que necesitan para crear virus peligrosos. Microsoft incluye las siguientes posibilidades de protección incorporadas que puede activar para protegerse de los virus de macro. Si utiliza Office 97, puede desactivar las macros de la siguiente manera:

1. Escoja **Opciones** del menú **Herramientas**.
2. Haga clic en la pestaña **General** del cuadro de diálogo y, a continuación, active la opción **Protección antivirus en macros**.

Si tiene Windows 2000, siga los siguientes pasos:

1. Escoja **Macro** del menú de **Herramientas**.
2. Haga clic en **Seguridad**.
3. Escoja el nivel de macro que prefiera (véase figura 23.2). La configuración **Alta** rechaza ejecutar macros excepto aquellas de fuentes conocidas.

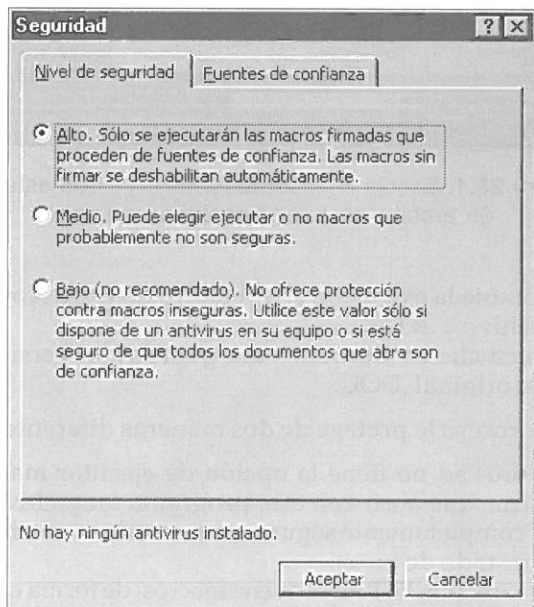


Figura 23.2. Use esta opción de seguridad para bloquear macros

Fíjese que mucha gente establece sus preferencias en los programas de gestión de correo electrónico de forma que abre los documentos adjuntos automáticamente. Naturalmente, si lo tiene así configurado, querrá cambiarlo de forma que pueda abrir documentos adjuntos manualmente. De esta forma, puede decidir qué documentos son seguros, como por ejemplo, aquellos de los que ha estado hablando con su colega. No habrá adjuntos que no espere recibir, aunque el mensaje provenga de alguien que conoce y en quien confía porque puede que ellos no lo hayan enviado.

23.3. Construya su propia macro

No es difícil crear una macro. Si hace una, puede diseñarla para que haga lo que más le convenga. Por ejemplo, puede hacer que se ejecute cada vez que se empiece a trabajar con Word. Esto es precisamente lo que hacen muchos autores de virus, entre ellos el autor del infame Melissa.

Para entender mejor cómo funcionan los virus, vamos a hacer una pequeña macro inofensiva que introduzca *Hola, Pedro* dentro de cada documento de Word nuevo cuando utilice Word.

Puede aprender el lenguaje de macros VBA para saber manejar sus cientos de comandos y características o puede ir por la vía rápida. Las macros se pueden grabar. Funciona de la siguiente manera: Arranca el grabador (**Herramientas, Macro, Grabar nueva macro**), hace que Word haga algo y, a continuación, detiene la grabación. Los comandos VBA se insertan en la macro que reproduce, guarda y simula aquello que queremos que haga Word.

En este ejemplo, he usado Word 2000 para hacer nuestra macro, pero el proceso de grabación es muy similar en cualquiera de las aplicaciones. Para crear una macro con Word 2000, elija **Herramientas, Macro, Grabar nueva macro** y llámela *Pedro*. Haga clic en **Aceptar** para comenzar el proceso de grabación de macro. Fíjese que le aparece una pequeña barra de herramientas con un botón **Pausa y Detener**.

Escriba *Hola, Pedro* y, a continuación, acepte. Haga clic sobre detener en la barra de herramientas para detener el proceso. Felicidades. Ha creado una Macro llamada *Pedro* y que escribe *Hola, Pedro*.

Pruebe la macro, presionando Alt-F8 (o escoja **Herramientas, Macro, Macros**). Localice *Pedro* en la lista y, a continua-

ción, haga doble clic sobre ella. La macro se ejecuta y escribe *Hola, Pedro* en el documento actual.

Echemos un vistazo al código de la Macro. Presione Alt-F8, haga clic sobre *Pedro* y, a continuación, sobre el botón **Editar**. Verá lo siguiente, con su nombre y fecha sustituyendo a los míos:

```
Sub Pedro()  
`  
` Pedro macro  
` Macro recorded 3/22/00 by Richard  
Mansfield  
    Selection.TypeText Text:="Hola, Pedro"  
    Selection.TypeParagraph  
End Sub
```

El comando *Selection* define la localización actual dentro del documento actual. Apunta a dónde aparece cualquier cosa escrita. La selección puede ser tanto una parte de texto seleccionado (destacado porque ha arrastrado allí el ratón) o, simplemente, el punto de inserción.

El comando *Typetext* hace lo que su nombre indica: escribe texto. Precisamente lo que se escribe es definido por la propiedad *Text*; en esta macro, el texto se define como *Hola, Pedro* y el final de la macro, la tecla Intro se activa con el comando *Typeparagraph*.

Cualquier línea de programa que comience por una sola comilla (') es simplemente un comentario. Le ayuda al programador a recordar cosas sobre el programa. Cuando se ejecuta la macro, se ignoran las líneas que comienzan por ('). Por lo tanto, puede escribir ahí lo que quiera o añadir nuevas líneas de comentarios. De forma similar, si quiere cambiar el mensaje, simplemente tiene que sustituir *Hola, Pedro* por lo que quiera que aparezca en el documento.

23.4. Disparo automático

Ahora que ha escrito su macro *Pedro*, ¿cómo hacer que funcione?. Como virus, es bastante sosito, simplemente escribe *Hola, Pedro*. No es malo y no quiere hacer daño. Simplemente quiere que se le reconozcan sus habilidades informáticas.

Durante algunas de sus operaciones, Word busca automáticamente ciertos nombres de macros. Si encuentra alguno de ellos, lleva a cabo instrucciones en la macro así

denominada. Por ejemplo, cada vez que abre un archivo .DOC de su disco duro, si hay una macro denominada AutoOpen, se ejecuta.

Si quiere que aparezca *Hola, Pedro* al principio de cada documento que abra, cambie el nombre de la macro de Pedro:

```
Sub Pedro ()
```

a

```
AutoOpen:
```

```
Sub AutoOpen()
```

Es todo lo que tiene que hacer. Ahora intente Archivo, Abrir. Localice un documento archivo .DOC, .TXT u otro tipo. Ábralo. En la parte superior, verá cómo Word escribe las palabras *Hola, Pedro* y, a continuación, desplácese a la siguiente línea (la orden *Selection.Typeparagraph* causa este comportamiento). Fíjese que en algunas versiones de Word puede tener que presionar F5 desde dentro del editor de VBA antes de que se ejecute la macro. Puede utilizar cualquiera de los siguiente autonombres especiales para una macro. Aquí tiene lo que hacen.

Nombre de la macro	Lo que hace
Autoclose	Cuando cierra un documento (Archivo, Guardar)
AutoExec	Cuando ejecuta Word
AutoExit	Cuando cierra Word (Archivo, Salir)
AutoNew	Cuando crea un nuevo documento (Archivo, Nuevo)
AutoOpen	Cuando abre un documento desde el disco duro (Archivo, Abrir)

Puede almacenar las automacros especiales donde guarda una macro normal; en la plantilla Normal1.dot, en otra plantilla o en un documento concreto.

Nota: Es aburrido pero puede invalidar las automacros presionando un tiempo la tecla shift. Por ejemplo si tiene una macro AutoOpen, presione shift mientras selecciona Archivo, Abrir. El documento se cargará sin abrir la macro AutoOpen.

23.5. ¿Es peligroso un mensaje normal de correo electrónico?

Ya sabe que no debe abrir adjuntos (archivos que vienen con el mensaje) que no espere recibir. Así es como se perpetua Melissa y virus similares. ¿Pero qué porcentaje de correo entrante no tiene adjuntos? ¿Es posible que un virus pueda viajar en un simple mensaje?

Por supuesto que sí. No es probable que le suceda, pero sí puede venir con correo. El problema es el scripting y, una vez más, objetos. En este caso, los objetos son applet o componentes de ActiveX.

Los script son pequeños programas, como las macros. Pero se supone que son limitados. Por ejemplo, al contrario de los que sucede con el lenguaje de programación de macros u otros, los lenguajes script son incapaces de eliminar archivos. JavaScript y VBScript son los dos lenguajes más conocidos actualmente y ninguno de ellos contiene comandos que puedan perjudicar su ordenador.

Sin embargo, aunque por sí mismos no pueden hacer daño, los script contienen objetos incluidos. Es muy similar a una manzana envenenada: la manzana en sí no es peligrosa pero puede transportar escondidas sustancias tóxicas. Mi consejo es que no coma nunca una manzana que le ofrezca una reina malvada y perversa.

23.6. Actualizaciones modulares

Otro desarrollo reciente que puede ser fuente de virus es la denominada actualización automática de software. Una parte del efecto de la programación orientada al objeto es que las aplicaciones actuales no vienen guardadas en un único y gran archivo. Son modulares (objetos) y sus variados componentes se almacenan en diferentes archivos. Las aplicaciones comparten archivos. Por ejemplo, Microsoft Outlook Express y Word disponen de la herramienta de corrección ortográfica. Sin embargo, no constan de correctores diferentes. Por el contrario, cada vez que cada uno de los programas usa el corrector, se activa el mismo módulo.

Ningún programa por muy complejo que sea está libre de fallos. Simplemente son muchas variables interaccionando entre ellas para chequear todos los movimientos que hacen. Por lo tanto, se vende software al público con defectos, pun-

tos flacos que sólo se hacen evidentes con el paso del tiempo. En el pasado, los distribuidores de software sacaban al mercado versiones nuevas de sus productos (v.1.1, 1.2, etc.) para solucionar problemas.

Sin embargo, Internet combinada con programas orientados al objeto ha hecho posible ofrecer actualizaciones como elementos descargables. Denominados paquetes de servicios, actualizaciones, y otras etiquetas, estos remiendos sustituyen módulos individuales (no hay que sustituir toda la aplicación, sólo los módulos que no funcionen bien). Este proceso puede automatizarse. Puede acordar tener una aplicación de Windows u otra que contacte automáticamente cada cierto tiempo con el sitio Web de actualización para saber si hay mejoras de fallos y si estos se pueden descargar automáticamente y ejecutarse. Este proceso garantiza que tiene la última y mejor versión de su sistema operativo y aplicaciones. El único problema es que puede introducir virus en su ordenador.

23.6.1. Protegerse de objetos infectados

Debería desactivar el proceso de Instalación de Microsoft Active si está trabajando con cualquier otro software. Puede que le salgan cuadros de diálogo pidiéndole permiso para actualizar los controladores de ActiveX (esto se hace para asegurar que tiene la última versión del software: mejoras en velocidad, nuevas propiedades o mejoras para solucionar fallos). Le recomiendo que rechace esto y desconecte la opción de actualizar si le aparece en alguna parte del sistema. Si lo desea, puede contactar con el sitio Web de Microsoft para ver si hay alguna actualización o parches que quiera instalar. Descárguelos manualmente. Puede confiar en estas actualizaciones de Microsoft porque se chequean mucho en busca de virus. Y es probable que lea en la prensa artículos sobre algún problema con actualizaciones automáticas (como, por ejemplo, virus escondidos). Lo que hay que evitar son las actualizaciones automáticas. No tiene por qué ser el primero de su edificio en tener las últimas versiones.

También puede visitar el sitio Web que contiene ActiveX u otros objetos. O quizás quiera recibir un mensaje de correo electrónico con un objeto incluido. Aunque esté firmado digitalmente, yo no probaría. Tome las siguientes medidas para protegerse:

1. Desactive cualquier opción de Descarga Firmada de Active X en todo su software.

2. Desactive Active Scripting (en Internet Explorer elija **Herramienta, Opciones de Internet**, vaya a la pestaña de **Seguridad** y haga clic **Personalizar nivel**).

Varias aplicaciones ofrecen diversas formas en las que puede establecer sus preferencias de seguridad. Abra el cuadro de diálogo de **Herramientas, Opciones**. En el programa de correo electrónico Outlook Express, puede especificar el nivel de seguridad. Entre otras configuraciones, puede estipular si se permite a los script o a los Controladores de ActiveX ejecutar mensajes de correo electrónico desde dentro de mensaje de correo HTML (puede no querer permitirlo).

Para evitar un ataque de virus, haga lo siguiente en Outlook Express:

1. Escoja **Herramientas, Opciones**.
2. Haga clic sobre la pestaña de **Seguridad**.
3. En la sección de **Zona de Seguridad** de la página de Seguridad, seleccione **Zona de Sitios Restringidos** (véase figura 23.3).

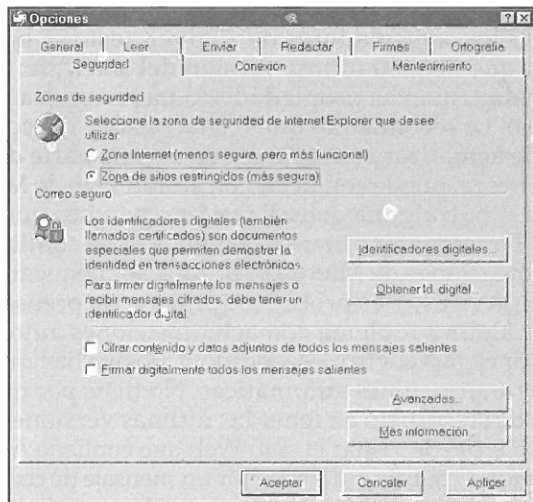


Figura 23.3. Escoja la opción de Sitios Restringidos para evitar ataques de script u otros objetos insertados en un mensaje de correo electrónico

Para protegerse contra ataques vía HTML (al visitar una página Web que contiene un objeto de ActiveX envenenado), ejecute Internet Explorer y siga los siguientes pasos:

1. Vaya a **Herramientas, Opciones de Internet**.
2. A continuación, vaya a la pestaña de **Seguridad**.
3. Haga clic sobre el botón de **Internet**.
4. Mueva la aguja hasta la configuración **Alta** (véase figura 23.4). Esto desactiva los cookies (datos almacenados en su disco duro por sitios Web), y también evita que se descarguen los controles de ActiveX. En mi opinión, Alta seguridad es desactivar demasiado. La configuración **Mediano** avisa de cualquier control sin firmar que esté a punto de ser descargado.

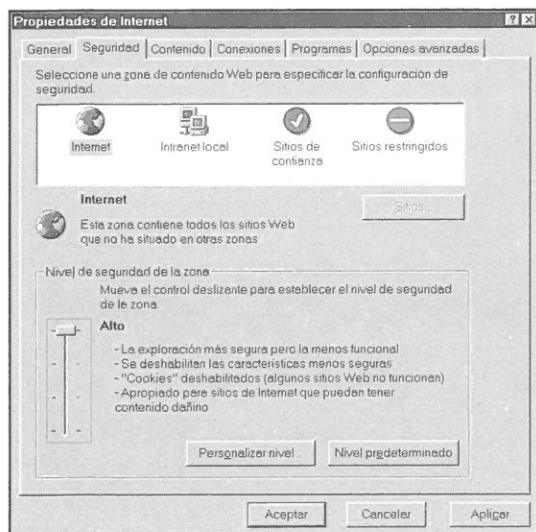


Figura 23.4. Utilice esta configuración para tener una protección máxima

Prevención, detección y eliminación

No puede vivir en este mundo y eliminar por completo todos los peligros. Bueno, puede intentarlo. Pero no debería ser tan paranoico como para no leer nunca correo electrónico, no visitar nunca sitios Web, guardar todos los movimientos que se realizan en su disco duro o ejecutar un antivirus diez veces al día. No tiene por qué ser el Howard Hughes de la informática, atrapado en su prisión virtual, guardando cada archivo borrado para realizar una inspección posterior por si al ordenador le pasa algo. Si hace todo esto, estaría al límite de la esquizofrenia.

Hay un camino racional intermedio entre la despreocupación total y la paranoia absoluta. Es el que le voy a recomendar en este capítulo. Siga los consejos que le doy a continuación y debería así poder evitar la mayoría de los virus y recuperarse pronto de los que no consiga mantener al margen.

24.1. Se exagera el miedo

Los virus no son esa gran amenaza que nos han hecho creer los medios de comunicación y los gobiernos. Yo lo comparo con los accidentes de tráfico. Son realmente impresionantes y, por eso, los periódicos locales y regionales están llenos de historias sobre camiones volcados, derramamientos de combustible, tragedias de accidentes de monovolúmenes y de autobuses. Pero, a pesar de esta fascinante atención de la prensa diaria a estos accidentes, no es muy frecuente ver uno en la realidad. Puede irse de vacaciones, cruzarse con más de 100.000 coches mientras se va a la pla-

ya y, probablemente, no vea ni un solo accidente. De hecho, puede vivir el resto de su vida sin tener ningún accidente serio.

USA Today y otras revistas populares en los Estados Unidos suelen publicar estremecedoras historias de virus cada semana, pero, ¿conoce a alguien cuya máquina fuera realmente atacada por un virus? Probablemente, no. De hecho, las posibilidades de que experimente uno a lo largo de su vida son bastante escasas. Tan escasas que casi siempre violó las seis normas que le voy a indicar en este capítulo.

Si es el responsable del sistema informático de una gran organización, seguramente no estará de acuerdo en absoluto con mi alegre despreocupación de los virus en este capítulo. El problema en ese caso puede ser muy serio porque es el protector del sistema y su puesto de trabajo peligra si triunfa un ataque en su sistema. Además, tiene la carga adicional de intentar enseñar a un grupo de gente a no abrir adjuntos inesperados de correo electrónico, aunque el mensaje lo envíe un conocido.

Los virus del tipo Melissa pueden ser una seria amenaza para la mayoría de las empresas. Mis condolencias. Sin embargo, aparte de las advertencias y sugerencias del capítulo anterior, sólo puedo hablar por mi propia experiencia en los últimos veinte años y por las experiencias que he leído de otros. Estas demuestran que el ataque de virus a usuarios particulares y a empresas es raro y, la mayoría de las veces, no causan tanto daño. Si aún así está preocupado, siga estas normas de sentido común que aparecen a continuación y reducirá casi a cero la posibilidad de la infección de un virus.

1. Si tiene un mensaje de correo electrónico con un adjunto, no abra este último. Hasta un documento aparentemente inocente .DOC puede tener un virus que infectaría su sistema en el instante en que abriera el documento. En vez de eso, envíele un mensaje a la persona que lo remitió o llámela por teléfono. Verifique que ha sido quien le envió el adjunto y que no es un virus del tipo Melissa que se auto envían automáticamente a todas las personas que aparecen en el libro de direcciones.
2. No baje archivos de la mayoría de los sitios de Internet. Se puede confiar en los archivos disponibles en Microsoft, Norton (antivirus) y otros sitios conocidos. Estos archivos se analizan cuidadosamen-

te para asegurar que no llevan ningún virus. Sitios como ZDNet (conocida fuente de programas shareware) o AOL se preocupan de escanear una y otra vez el software que ofrecen.

3. Elimine los archivos adjuntos de un mensaje de correo electrónico si este ha sido enviado por alguien desconocido. No intente nunca ejecutar o visualizar esos archivos.
4. Desactive las macros. Vaya a Herramienta, Opciones o Herramientas, Menús de macro para localizar la configuración de la seguridad de macros y sus aplicaciones principales (procesador de textos, hojas de cálculo, bases de datos, programador de tareas, etc.). Configure la seguridad en el nivel más alto de forma que sólo se puedan ejecutar las macros de fuentes conocidas o registradas.
5. Escanee el software nuevo. Vaya a <http://www.mcafee.com> o a http://www.symantec.com/region/reg_eu/product/navbrochure para obtener información sobre los dos programas de antivirus más famosos. Cuando alguien le da un archivo nuevo, le presta un disquete o consigue software por otras fuentes, escanéelo utilizando las utilidades de McAfee o Norton. Y asegúrese de mantener estos productos actualizados. Contacte con sus sitios Web con regularidad para actualizar la base de datos de su antivirus y asegúrese de que chequea las últimas versiones de virus. Recuerde que cada día se escriben y se sueltan en la red docenas de virus nuevos. Las posibilidades de que le llegue uno son bastante escasas, pero de todas formas saldrá adelante, ¿verdad?. McAfee mantiene una base de datos con más de 45.000 virus. Aunque hay una corriente continua de virus nuevos, es más probable que entre un virus antiguo que uno de estos nuevos.

***Nota:** Alguno expertos no están de acuerdo con este consejo que doy aquí. Afirman que si tiene un programa antivirus esté activo constantemente, es más probable que entren virus nuevos porque el programa detectaría, bloquearía y eliminaría los existentes.*

6. Considere instalar un antivirus que esté funcionando continuamente. Algunos pueden hacerlo, siempre que

su ordenador está encendido. Esto ralentiza su sistema un poco, pero si ha tenido serios problemas, puede ser la manera más rápida para detectar y solucionar el problema. La mayoría de nosotros no necesitamos este nivel de protección, pero si tiene importante información irremplazable, un elevado número de software para ejecutar o su organización es vulnerable a un ataque de virus significativo, debería usar este tipo de antivirus.

Todo lo que puedo decir es que, durante los doce años que he trabajado para grandes empresas editoriales que estaban absolutamente informatizadas y durante los veinte años que he estado descargando software y programando, ni yo ni esas empresas hemos tenido virus. Y las estadísticas dicen que tanto personas como organizaciones que sí han sufrido ataques, se recuperan rápidamente. Sin embargo, si está teniendo problemas con los virus, tome las medidas anteriores para defenderse.

24.2. Mi confesión

Me gusta descargar archivos de música .MID y .MP3 de diversos sitios de Internet. Con frecuencia tengo que abrir archivos (archivos .DOC) que recibo como adjuntos en mensaje de editores, sin que ellos me hayan avisado previamente del envío.

Uso muchas macros que me ayudan con tareas habituales (como el formato de encabezados) en el procesador de Word para Windows. No me preocupo de escanear el software que compro. Doy por hecho que los juegos que compro de Electronic Arts están limpios. No utilizo un antivirus residente en memoria. De hecho, no uso ningún antivirus.

Debo confesar es que la mayoría de las veces no sigo las normas que mencioné anteriormente (pero sí evitaría abrir un archivo adjunto a un mensaje de correo de alguien que no conozco).

En vez de una protección antivirus, que considero más incordiante que útil dadas las escasas posibilidades de infección, he tomado otra medida. Si alguna vez entra un virus en mi ordenador, el daño sería mínimo. Tengo todas mis aplicaciones, incluido Windows, grabadas en CD. Compré las aplicaciones, así que tengo los discos. Puedo reinstalar

fácilmente mi procesador de textos, mi navegador de Internet y otras aplicaciones. También puedo reinstalar el sistema operativo si es necesario (pocos virus son tan crueles como para eliminar el sistema operativo).

Y, lo más importante, *hago copias de seguridad de mis archivos todos los días.*

24.2.1. También Michael Jackson se resfría

Recuerde que ni siquiera los antivirus que escanean continuamente pueden evitar ataques de virus. Por lo tanto es imposible protegerse de todas las infecciones de virus. Es tan probable que coja un catarro con frecuencia (aunque lleve una mascarilla protectora como Michael Jackson) como que sufra un virus informático en algún momento de su carrera informática, sin importar las precauciones que tome. Recuerde que yo nunca he tenido virus y he estado usando el ordenador cuatro horas diarias desde que me compré mi primer ordenador en 1980.

Para resumir todo el tema de la protección antivirus: no son habituales y es probable que jamás tenga uno. Pero si le preocupa, la solución es simple: *Haga copias de seguridad de sus archivos.*

24.2.2. La mejor protección

Mejor táctica que cualquier antivirus es hacer copias de seguridad de sus archivos. Tiene que hacer copias de sus archivos de datos (archivos .DOC y otros creados mientras trabaja con sus aplicaciones). El peligro real de un ataque de virus no es que vaya a corromper su procesador de textos, ya que siempre puede reinstalarlo en unos minutos si tiene los CD originales. El peligro de estos virus es que pueden eliminar o estropear información irremplazable (todos los mensajes de correo electrónico recibidos y enviados, cartas escritas, hojas de cálculo elaboradas, todos sus trabajos escolares). En otras palabras, puede perderse para siempre el resultado de toda su actividad informática.

Para prevenir esta catástrofe, deben hacerse con regularidad copias de seguridad de los archivos. Emplee una utilidad para hacer copias de seguridad y guarde sus archivos con frecuencia. De esta manera, no perderá todo lo que tiene a causa de un virus. Hacer esto lleva poco tiempo y es, con diferencia, la mejor protección contra los virus. La primera

vez puede llevar diez minutos o una hora, dependiendo de su productividad. Pero una vez que haya hecho copia de seguridad de todo, las copias posteriores las realizará en unos segundos porque lo que tiene que hacer es guardar archivos nuevos o modificados. Windows y su utilidad de copias de seguridad guardan información sobre los archivos que han sido ya guardados (archivados), y consecuentemente, saben los que hay que guardar.

El daño a la información es inevitable

Aunque nunca he tenido un virus, sí se me han estropeado datos a lo largo de mis veinte años frente al ordenador. He tenido problemas con discos duros que funcionaban mal, disquetes que perdí o un disco Zip que funcionaba mal. Sin querer, he sobrescrito archivos. Por ejemplo, una vez copié todo un capítulo que había terminado para un libro. Lo utilicé como plantilla para comenzar a escribir el siguiente. Cambié el número de capítulo, le di otro nombre y, a continuación, borré el contenido, es decir, todo el texto del documento. A continuación, guardé el archivo. Lo que hice mal fue cambiar el nombre de archivo, de forma que, en un momento de locura, borré todo el trabajo de muchos días sustituyendo el documento válido por una nueva versión vacía de contenido. Este tipo de pérdida de información me enseñó a hacer copias todos los días. Por supuesto, estas copias rutinarias son su mejor protección ante virus que puedan destruir información. Sinceramente, no creo que los antivirus sean muy prácticos. Pero también pienso que es una tontería que no haga copia de seguridad en su disco duro habitualmente. No tanto por el peligro ilusorio que puedan suponer los virus, sino porque el verdadero peligro que supone la pérdida de información causada por soportes que no funcionan bien y meteduras de pata como la eliminación del capítulo que acabo de describir.

24.3. Lo que hacen los programas antivirus

Si, a pesar de todo, es una de esas personas que insiste en mantener los virus alejados de su ordenador, hay varias formas de defensa que puede usar para reforzar su sistema.

24.3.1. Escanear las cadenas de números

Todos los datos y programas informáticos están compuestos de bytes individuales. Cada byte contiene un número entre 0 y 255. Si pudiera ver las celdas individuales de memoria de su ordenador (o de su disco duro), vería una cadena de números. Por ejemplo, vería algo así: 2 230 231 66 98 149,

Las letras del abecedario están codificadas de forma que el número 65 se corresponde con la *A* mayúscula, 66 con *B*, etc. Las minúsculas aparecen a continuación, empezando la *a* en el número 97. Mi nombre, Richard, tendría este aspecto en la memoria del ordenador: 82 105 99 104 97 114 100. Este es el código ASCII utilizado en algunos tipos de cifrado informático, como se explica anteriormente en este libro.

De la misma forma, las instrucciones de programas informáticos tienen sus propios valores numéricos. Por lo tanto, si escanea un disquete, un disco duro, la memoria RAM del ordenador o cualquier otro elemento de almacenamiento, puede buscar cadenas concretas (grupos) de datos. Cada vez que mi nombre aparezca en un documento, por ejemplo, aparecerá la cadena 82 105 99 104 97 114 100.

Por supuesto, los autores de virus no son tan torpes como para poner su nombre y número de teléfono dentro de estos. Pero tienen que incluir líneas de programación como decirle al ordenador que despliegue el mensaje «Nos vamos de vacaciones», que borre archivos o cualquier otra cosa. Una vez que se descubre un virus, pueden identificarse sus cadenas características. A continuación, pueden ser escaneadas por un antivirus que va leyendo el código de los programas o los datos en un disco o en la memoria RAM buscando estas cadenas. Los ordenadores son muy buenos y muy rápidos escaneando. Por ejemplo, si está leyendo un documento en su procesador de textos y selecciona la opción Edición, Buscar, el ordenador puede buscar instantáneamente en todo el documento para localizar la cadena dada.

24.3.2. Escondiéndose en el horizonte: el virus contraataca

Los antivirus tienen algunos fallos. La desventaja principal es que ya se ha producido la infección en el momento en que detectan el virus. Además, el autor del virus puede

cubrir el virus, es decir, ocultar las cadenas distintivas. Las técnicas de autoencriptado permiten que el virus genere automáticamente una versión cifrada de sí mismo y la guarde en cada programa que infecta. La clave para el cifrado puede ser aleatoria (basada, por ejemplo, en el tiempo presente) y guardada dentro del virus encriptado. El resultado último es que cada nueva copia del virus parecerá diferente durante el proceso de escaneo. El único fallo de este sistema es que el virus tiene que descifrarse y volver a su forma original para ejecutarse. Esto significa que no se puede cifrar el código de desciframiento, y, consecuentemente, este sí será detectado durante el escaneo ya que no puede variar (no se puede cifrar). Parece que todo tiene un talón de Aquiles.

Otra medida que toman los autores de virus para evitar la detección durante el escaneo es la mutación de sus virus. Similar al cifrado, la mutación significa cambiar algunos de los contenidos de los virus. Pero, con la mutación, el cambio es permanente y cada nuevo virus engendrado difiere de su padre. ¿Cómo puede hacer un virus esto sin perjudicar su capacidad para infectar, comprobar sus condiciones y dejar su carga? La respuesta es que existen diversas maneras de realizar una misma función en un programa. Por ejemplo, WordBasic tiene distintos comandos sinónimos que puede utilizar para que un mensaje aparezca en medio del texto de un documento activo. Del mismo modo, en el lenguaje ensamblador, podría cargar en un registro el valor cinco (LDX 5) o incrementar ese registro X cinco veces (INX INX INX INX INX); en ambos casos, se obtiene el mismo resultado, pero utilizando un código diferentes.

A los creativos autores de virus no les faltan recursos para inventar otras formas de ocultar sus obras de arte de los patrones de escaneo. Los programadores pueden hacer que la ejecución del virus mueva bloques de código dentro del programa. Esto significa que, por ejemplo, podría mover partes completas de un programa, mientras ajusta las direcciones a las que se tiene que mover. Piense en esta táctica como si fuese un cartero que hace una lista de las direcciones a las que tiene que ir y modifica el orden todos los días. Puede introducir un elemento de aleatoriedad en el orden en el que se listan las direcciones y, por lo tanto, en el orden en el que el programa se va a desplazar.

Todos los lenguajes informáticos tienen la capacidad de saltar a cualquier dirección y recordar desde dónde lo hicieron al toparse con una instrucción de *retorno*. Esto significa

que, durante la ejecución del virus, este puede ir saltando a cualquier lugar dentro de su código, pero volver automáticamente a ejecutar una instrucción de salto diferente (en una lista con este tipo de instrucciones). Las implicaciones de esta característica de *salto/retorno* es que se pueden mover partes del programa en cualquier orden como si estuviera reajustando las piezas de un rompecabezas constantemente. Evidentemente, este tipo de técnicas despistarían a cualquier antivirus que buscara patrones predecibles durante el escaneo.

24.3.3. La técnica Mata Hari: detección mediante atracción

Una interesante técnica antivirus es la que envía un programa a su disco duro que tiene como único propósito mover sus caderas para atraer a los virus. El programa pretende que el virus lo invada y lo infecte. Su único propósito es seducir a los virus (existen ciertos tipos especiales de acciones de los programas que atraen a los virus). Este programa señuelo se autocontrola para ver si ha sido modificado (por ejemplo, si es más grande). Si ha cambiado, alerta al software antivirus.

En relación con este enfoque de señuelo está la detección por actividad. Los virus tienen que dar ciertos pasos para llevar a cabo sus actividades, entre los cuales está abrir archivos .EXE; esta extensión significa «ejecutable» y es la extensión propia de la mayoría de los archivos de aplicaciones. Winword.exe es un ejemplo. La función de estos archivos .EXE es ser ejecutados, no ser abiertos para leer o escribir. Utilizando un símil diríamos que el comportamiento propio del motor de un coche es encenderse; un comportamiento impropio es que abra el capó y desatornille el filtro del aire. Si se lee o escribe un archivo ejecutable, va a pasar algo muy, muy malo.

Los virus pueden hacer una gran variedad de cosas, siempre llamando la atención. Por ejemplo, la mayoría de los virus deben leer el sector de arranque y escribir nuevas órdenes en él. Alternativamente, otros virus escriben cosas nuevas en archivos .EXE u otros archivos ejecutables, algo que nunca debería ocurrir. Una de las reglas más antiguas de la programación es no escribir nunca un código que se automodifica.

24.3.3.1. Es fácil mantener el tamaño de un archivo

Las estrategias simples de detección, como mirar si algún archivo .EXE ha modificado su tamaño, no son ya útiles. A un virus no le importa hacer daño a alguno de sus programas. Para evitar modificar el tamaño de un archivo, el virus puede simplemente sustituir el código que quiera en un archivo .EXE para así mantener el tamaño preciso de ese archivo, dañando probablemente una o varias funciones de la aplicación durante el proceso. Recuerde que un virus puede tener tan sólo 100 bytes y una versión de Winword.exe (Word 2000) tiene un tamaño de 8.799.232 bytes. No es difícil introducir un pequeño virus dentro de Word sin modificar el tamaño de sus archivos.

24.3.4. Enganchar interrupciones

Formatear el disco duro es otra acción muy infrecuente en la informática común, sin mencionar el elevado daño que causa a los programas y datos que contiene. Algunos programas de detección de virus redireccionan las *interrupciones* (direcciones que pueden usar las aplicaciones para llevar a cabo diferentes tareas) de forma que las peticiones de formateo se envían a través del software de detección (por ejemplo, el DOS de Microsoft incluye el grupo completo de interrupciones). Cuando un programa acciona una interrupción, esta realiza su trabajo, que puede ser leer directorios, archivos y, como puede imaginar, algo tan infrecuente como reventar un disco formateándolo.

Un programa detector de virus puede interceptar todas las peticiones de formateo y avisarle que se está llevando a cabo un intento de formateo. A continuación, el programa le pregunta si de verdad, de verdad quiere formatear. Desgraciadamente, el virus puede esquivar las interrupciones predeterminadas y causar daño directamente. También, los astutos autores de virus pueden analizar los vectores de interrupciones (las direcciones de destino) para ver si, de hecho, se ha redireccionado alguna interrupción. De forma que la técnica de redireccionamiento funciona, sólo a veces, de forma efectiva para prevenir el formateo u otras acciones perjudiciales.

24.3.5. Escaneo típico para detectar cambios en los archivos

El procedimiento típico del escaneo antivirus implica normalmente chequear sus aplicaciones (.EXE y otros archi-

vos ejecutables) para ver si han cambiado. Hay varios cambios posibles que un virus puede realizar cuando invade un archivo: el tamaño de este, su hora y fecha, y el checksum. Es muy fácil para un virus mantener el tamaño, y la fecha y hora del archivo. Sin embargo, para un virus es relativamente difícil mantener el mismo checksum después de haber sobrescrito parte del programa o todo.

¿Qué es un checksum? Recuerde que los códigos informáticos (programas) pueden ir de 0 a 255 en cada byte de un programa. Algunos códigos le dan instrucciones al ordenador para que muestre texto, otros guardan datos y otros computan (sumar números, revisiones ortográficas, etc.) Pero, hagan lo que hagan, la programación informática en su nivel más bajo es una lista de números que van de 0 a 255. Para ilustrar la idea de checksum, supondremos que un programador escribió una utilidad con un pequeño archivo de guardar que tenía diez instrucciones de largo. Tendría esta apariencia: 27, 157, 2, 88, 240, 240, 8, 99, 201, 84. Para obtener el checksum de un archivo, se suman los valores de código del archivo: $27 + 157 + 2 + 88 + 240 + 240 + 8 + 99 + 201 + 84$.

El checksum de nuestra utilidad imaginaria es 1146 (en la realidad, la utilidad sería muchísimo más larga y, consecuentemente, también lo sería el checksum). Ahora, el virus sustituye alguno de esos códigos. Hay pocas posibilidades de que la suma tenga el mismo resultado que el código original. Por eso, aunque el tamaño (número de bytes) del archivo infectado sea el mismo, el checksum varía.

¿Tiene inconvenientes este enfoque? Por supuesto. Nada es perfecto. ¿Recuerda que Word tiene un tamaño de 8.799.232 bytes? Hasta a los ordenadores, rápidos como son, les lleva un rato sumar hasta nueve millones de números. Las aplicaciones actuales son muy largas y cada vez que se escanean, hay que calcular el checksum. Además, algunos programas de escaneo dejan un montón de pequeños archivos con resultados de la suma esparcidos por el disco duro y otros llegan a sumar el resultado a los archivos .EXE de la aplicación, cambiándose así el tamaño del archivo.

***Nota:** Si está profunda y realmente preocupado por los virus, y tiene un montón de conocimientos técnicos (por ejemplo, entiende el lenguaje ensamblador/desensamblador) hay sitios de Internet donde puede obtener información detallada. Por ejemplo, encontrará detalles minuciosos sobre una*

amplia variedad de virus, entre ellos listados comentados sus código en el sitio Web www.fc.net/phrack/under.html. Este sitio de la revista Phrack tiene archivos de varias newsletters y zines, entre ellos 40Hex, una de las newsletter más famosas de todos los tiempos (léida ávidamente por autores de virus y programadores de escaneo antivirus). Un desensamblado es una lista legible de instrucciones de programa. A un programa de utilidades denominado desensamblador se le puede dar un trozo de código de programación (por ejemplo, un virus) para que traduzca el código máquina en instrucciones de lenguaje ensamblador, comprensibles para usuarios avanzados.

El sitio Phrack incluye : 40Hex, Anarchy 'N' E.xplosives, The Art of Technology Digest, Activist Times Inc., The BIOC Files, The Cult of the Dead Cow, Chalisti, Digital Free Press, Freakers Bureau Incorporated, Freedom, Informatik, Chaos Digest, The Legion of Doom Technical Journals, Legions of Lucifer, Miscellaneous Underground Files, N.A.R. C Newsletter, The New Fone Express, Network Information Access, National Security Anarchists, Phantasy magazine, PHUN magazine, Pirate magazine, The Syndicate Report, United Phreakers Incorporated Newsletter, Vindicator Publiotions y The WorldView. Un mundo de revistas alternativas.

En resumidas cuentas: es improbable que un virus dañe los datos de su ordenador, pero si le preocupa, este capítulo le ofrece distintas medidas que puede tomar para proteger su máquina.

Índice alfabético

A

- Activar (un virus), 261
- Active X, 288
 - Controladores de Active X, 289
- Actualización modular, 288
- Administrador de sistema, 197
- Agencia Nacional de Seguridad de Estados Unidos, 164, 170
- Agente, 250
- Agujero de seguridad, 260
- Ajustar línea de código, 241
- Alerta de seguridad
- Alexa, 117
- Algoritmo, 157
 - Algoritmo generador de números aleatorios, 229
- Alice, 220
- Alt.2600.hackerz, 85
- Alt.2600.warez, 64
- Alt.sex (grupo de noticias), 266
- Amazon, 51, 117
- Análisis de la frecuencia columnar, 230
- Anonymizer, 114
- Anti-hacker, 65
- Antivirus, 277, 298
- AOL, 54, 81, 106, 267
- Archivo
 - Archivo .CR2
 - Archivo .CRP
 - Archivo .JPG, 275
 - Archivo .MID, 296
 - Archivo .MP3, 275

- Archivo .RTF, 283
- Archivo .TMP, 197
- Archivo .TXT, 238, 275
- Archivo .VBE, 279
- Archivo adjunto, 277, 288
- Archivo cebo, 66
- Archivo de datos, 274
- Archivo de datos y acciones, 281
- Archivo ejecutable .EXE, 273, 301
- Archivo list.DOC, 271
- Archivo. DOC, 157, 258, 283
- Archivos temporales de Internet, 81
- Bajar archivos de Internet, 294
- Compartir archivos e impresoras, 34, 83
- Guardar archivo modificado, 298
- Archivo fuente, 236
- ASC, 231
- ASCII, 78, 158, 229, 299
- Autenticación digital, 189
- Autenticar, 185
- Auto reproducirse, 255
- AvenueA, 104

B

- Base de datos, 261
- BBS, 160, 246, 273
- Bit, 155, 168
- Bloc de notas, 256, 274
- Bloque de código, 300
- Bloqueo de ordenador, 251
- Blowfish, 115
- Bob, 220
- Bot, 51
- Bucle, 230
- Bucle principal, 236
- Buffer, 202
- By Kwyjibo, 272
- Byte, 155, 299

C

- Caché temporal, 81
- Carácter, 126, 148

- Carácter cifrado, 131, 163
- Carácter clave, 142
- Carnivore, 108
- Carpeta TEMP, 197
- CD regrabable, 257
- CD-ROM, 109
- Cero numérico, 157
- CGI, 94
- Checksum, 303
 - Modificación en archivo, 303
- Chips de memoria, 256
- Ciber autenticación, 186
- Ciber réplica, 106
- Ciberespacio, 58, 115
- Ciberespionaje, 104
- Ciberguerra, 246
- Ciberperfil, 268
- Cibervandalismo, 213
- Cifrado submolecular, 213
- Cifrador, 239
- Cifrar, 125
- Clave, 72, 115, 155, 210
 - Centro de distribución de claves, 175
 - Clave aleatoria, 226
 - Clave cifrada, 76
 - Clave criptográfica, 170
 - Clave de recuperación, 196
 - Clave de sesión, 175
 - Clave de usuario, 232
 - Clave inmediata, 158
 - Clave interna, 241
 - Clave memorizada, 226
 - Clave privada, 173, 180
 - Clave pública, 173, 180
 - Clave secreta, 134
 - Clave temporal, 175
 - Clave única, 175
 - Generar una clave, 226
 - Intercambio de claves, 183, 219
 - Par de claves, 180
 - Sistema de claves, 181
 - Transferencia de claves, 174
- Clave del registro de Windows, 272
- Cloaking, 112
- Codificar, 125

- Código ANSI, 148, 155
- Código de arranque, 256
- Código de programa, 275
- Código ejecutable, 257
- Código fuente, 240
- Código numérico, 148
- Código secreto, 259
- Colossus, 164
- Comercio electrónico, 47, 185
- Comisión Federal de Comercio
- Compañías .com, 107
- Compresión, 129
 - Compresión por permutación, 168
- Conexión de banda ancha, 29, 89
- Confusión, 166
- Contador, 252
- Contrainoculación, 252
- Contraseña, 50, 74, 134, 158
- Cookie, 98
 - Permitir que se almacenen cookies en el equipo, 99
 - Personalizar nivel, 98
 - Seguridad, 98
- Copia de seguridad, 265, 297
- Correo electrónico, 49, 79, 109, 288
 - Bloqueo de correo electrónico, 253
 - Correo basura, 50, 105
 - Correo entrante, 55
 - Cuentas de correo electrónico desechables, 111
 - Cuentas de correo gratuitas, 111
 - Dirección falsa, cuenta falsa, 55
 - Disfrazar la dirección de correo electrónico, 53
 - Enviar correo electrónico de forma anónima, 49
 - Libro de direcciones de correo electrónico, 122
 - Mensaje de correo, 269
 - Mensaje entrante, 53
 - Omitir la dirección de correo electrónico, 52
 - Reenvío anónimo, remailer, 112
 - Sistemas de gestión de correo electrónico, 266
- Cortafuegos, 37, 75, 79
- Counterpane Internet Security, 70
- Cracker, 45, 59, 85
- Craquear, 146
- Craquear sistemas telefónicos, 41
- Craqueo de cifrado, 63
- Crawler, 51

- Criptoanálisis, 127
- Criptoanalista, 141, 163
- Criptografía, 114, 125
 - Criptografía atómica, 213
 - Criptografía cuántica, 219
 - Criptografía informática, 147
 - Sistema criptográfico, criptológico, 141, 225
 - Sistema de cifrado, 143
- Criptología, 130, 145
 - Criptología computacional, 163
 - Criptología electrónica, 164
- Cuanto, 217
 - Estado de, 218
- Cubit, 217

D

- Daemon dialer, 39
- Dato punto flotante, 232
- Datos electrónicos, 247
- DEF CON 00, 47
- Denegación de servicio, DOS, 86
- Depositar carga, 263
- DES, 163, 191
- Descarga firmada, 289
 - Desactivar, 289
- Descifrar, 127, 138
- Difusión, 166
- Dirección IP, 31, 71, 80
 - Dirección IP externa, 80
 - Dirección permanente IP, 30
- Disco A:, 256
- Disco zip, 257
- Disquete, 257
- Distorsión, 149
- Distribución cuántica, 223
- División en bloques, 129
- DoubleClick, 104
- Duplicarse (un virus), 251

E

- EarthLink, 89,108
- Efecto 2000, 245, 261

- Ejecutarse, 256
- Electrón, 215
- Emisor, 155, 174
- Empresa de seguridad informática, 68
- Encryptor, 239
 - 1000.crp, 240
 - Encrypt, 240
 - Qwer, 240
 - Shortcut, 240
 - Shortcut key, 240
- ENIAC, 164
- Enigma, 163
- EPA, 189
- Epic, 112
- Escaneo de seguridad, 94, 300
- ESN, 43
- Espacio en blanco, 134, 157
- Estado latente, 249
- Estándar, 166
- Esteganografía, 125
- Ethernet, 267
- Eve, 220
- Excite@Home, 89
- Expansión, 129
 - Expansión por permutación, 168
- Exponenciación, 181
- Extensión de virus, 257

F

- Fabricantes de software, 264
- Factor, 179
- Factorizar, 179
- FakeAddr, 54
- FedEx, 188
- Ferret, utilidades, 56, 60
- Fibra óptica, 219
- Filtros, 54, 109
- Firma digital, 107, 186
- Firma electrónica, 185
- Fisgón, 70
- Formulario, 236
- Fotones, 213
 - Fotones enlazados, 220
- Frecuencia, 128

- Análisis de frecuencias, 151
- Frecuencia de caracteres, 142
- Nivel de frecuencia, 159
- Patrón de frecuencia, 231
- Tabla de frecuencias, 128
- Free Security Scan, 95
- Freedom, 114
- Fujitsu, 75

G

- Ghost Mail, 49
- Gibson, Steve, 37
- Gibson Research Corporation, 34
- Google, 49
- Grabador de tonos, 43
- Grupos de noticias, 49
- GUID, 117, 266
- Gybson, William, 94

H

- Hackear, 271
- Hacker, 29, 45, 64, 247
- Hacker Central, 63
 - Grupo de noticias, 63
- HackerWhacker, 94
- Hipervínculo, 30, 95, 105
- Honeypot, 97
- Host, 250
- HTML, 290
- HTTP, 71
- Huevo de pascua, 258
- Hushmail, 111, 243

I

- ID, 63, 112, 267
- Idcide, 104
- Independiente de plataforma (virus), 279
- Infección, 273
- Infectar, 265
- Infinito, 216
- Infinitos monos, 153, 222

- Infraestructura informática, 245
- Infracore's Sanitizer, 123
- Ingeniería social, 40, 58, 73
 - Invertir la ingeniería social, 73
- Input, 68
- Institute of Electrical and Electronic Engineers
- Intercambio electrónico de fondos, 247
- Interfaz de 10 dedos, 67
- Interfaz de usuario, 147
- Internet Explorer, 100, 291
 - Nivel de seguridad alto, 291
 - Nivel de seguridad bajo, 276
 - Nivel de seguridad mediano, 291
- Internet Sky, 274
- Interrupción, 302
- Intranet, 282
- Intruso, 131
- IPO, 189

J

- Java, 168
- JavaScript, 288

K

- Kerckhoffs, Auguste 140
 - Alineación, 142
 - Antitabla, 141
 - Desciframiento por superposición, 142
 - La Cryptographie Militaire*, 140
 - Superposición, 140

L

- LAN, 33, 85
- Larvae, 47
- Lenguaje binario, 221
- Lenguaje C, 168
- Lenguaje ensamblador, 304
- Lenguaje natural, 136, 157
- León, 133, 136
 - Alfabetos cambiantes, 142
 - Códigos polialfabéticos, 135

- Sustitución polialfabética, 133, 142
- Tabla de sustitución polialfabética, 134
- Ley del Comercio Digital del Milenio, 188
- Localización de carácter, 230
- Lógica booleana, 181
- LOVELETTER, 274
- Lucent, 266
- Lucifer, 165

M

- Macro, 269, 284
 - Construir una macro, 285
 - Ejecutar macros de fuentes desconocidas, 284
 - Grabar nueva macro, 285
 - Lenguaje de macro, 269, 282
 - Macro Autoclose, 287
 - Macro AutoExit, 287
 - Macro AutoNew, 287
 - Macro AutoOpen, 287
 - Menú de macro, 295
 - Protección antivirus en macros, 284
 - Selection, 286
 - Typeparagraph, 286
 - Typetext, 286
 - WordBasic, 282
- Mafiaboys, 87
- Mask, 236
- MatchLogic, 104
- Mbps, 84
- McAfee, 89, 295
- McAfee Software's Guard Dog, 101
- Media Access Control, 36
- Memoria RAM, 299
- Memoria volátil, 252
- Mensaje, 153
 - Mensaje cifrado, 125
 - Mensaje original, 142
- Microsoft NetMeeting, 31
- MIN, 43
- MIT, 245
- Mitenik, Kevin, 46
- Mixmaster, 113
- Mockingbird, 58
- Modelo criptográfico, 131

Modelo de cifrado, 130
Módem cable, 39, 83
Morris, Robert Tappan, 251
Motor de búsqueda, 55
MSN, 81
Muerto viviente, 47

N

NASA, 248
NetBIOS, 35
Netscape, 64, 100
Newsletter, 304
Nivel atómico, 216
Nivel de aplicaciones, 78
Nivel de enlace, 78
Nivel de permiso de acceso, 194
Nivel de red, 78
Nivel de seguridad (de un sistema), 66
Nivel de sesión, 78
Nivel de transporte, 78
Nivel de usuario, 57
 Cuenta de superusuario, 260
 Nivel de acceso bajo, 57
 Superusuario, 57
No-repudiación, 187
Nombre de usuario, 98, 271
Norton Internet Security 2000, 93, 101
Nulls, 128, 169
Número aleatorio, 221
 Lista pública de números aleatorios, 230
Número PIN, 160
Número primo, 179
 Tabla de números primos, 178
Números enteros, 177

O

Ocultar el virus, 300
 Versión cifrada, 300
Oficina Nacional de Estándares, 165
On Track's Data Erase, 123
Onda de probabilidad, 216
One time pad, 127, 225

Ordenador cuántico, 214
Organización Internacional de Estándares, 78
Outlook Express, 64
Output, 68

P

Paquetes de datos, 79
Paquetes de servicios, 289
Patrón, 153, 167
Pérdidas económicas, 269
Perfil personal, 104
Permutaciones, 129, 146, 173
Petrouts, Evangelos, 227
Phone Phreaks, 39
Phreaking, 40, 63
Ping, 86, 91
Plantilla Normal.dot, 269, 287
Polaridad, 215
Polimórfico, 279
Política de seguridad, 74
Pop3now
Potenciar, 181
Pretty Good Privacy, 243
Principio de incertidumbre de Heisenberg, 216
Privacidad, 85, 103
 Ley de la Privacidad en Comunicaciones Electrónicas, 42
 Política de privacidad, 107
 Privacidad de datos, 71
 Privacidad en Internet, 103
 Privacidad P3P, 111, 115
Privacy Companion, 104
Proceso comunicativo, 187
Profesionales IT, 235
Programa legal, 255
Programa receptor, 262
Programación orientada al objeto, 281
Programador, 261
Programador de virus, 257
Programador de Visual Basic, 241
Programas ET, 116
 Llamar a casa, 116, 264
Proporción lineal, 144
Protector de pantalla, 259
 Texto 3D, 259

- Tuberías 3D, 259
- Protocolo Diffie-Hellman, 226, 243
- Protocolo Tunneling de Nivel 2, 77
- Protocolos de Windows, 32
 - FTP, 33, 78
 - HTTP, 33
 - IPX/SPX, 32
 - NetBEUI, 32
 - TCP/IP, 33, 94, 193
- Proveedor de Servicio de Internet, 51, 108
- Publicación de Estándares de Procesamiento, Información Federal, 170
- Puerta trasera, 58, 166, 260
- Puerto, 29, 35, 84, 95
- Punto flaco (en la defensa), 263, 289
- PWS, 94

Q

- QE, 215

R

- RADIUS, 81
- Rand, 248
- Rango, 228
- Rat Dance, 62
- Real Jukebox, 268
- RealNetworks, 116
- Receptor, 155, 174
- Recompilar, 240
- Red de área local, 42
- Red de Área Privada, 77
- Red informática, 246
- Red Virtual Privada, 77
- Redireccionador, 118
- Regedit, 272
- Registro de Windows, 272
- Reinstalación del sistema operativo, 265
- Residente, 252
- RND, 228
- Robo de contraseña, 263
- Robo de identidad, 190
- ROP, 225, 238

Rotación, 168
Rotar la clave, 237
Rotor, 163
RSA, 173, 186, 191, 214

S

Scatha y Glaurung, 250
Script, 288
Sector de arranque, 256
Seguridad de red, 67
Servidor Personal de Páginas Web, 34
Servidor proxy, 81
Shareware, 273
Shields Up!, 34, 97
 Probe My Ports!, 35
 Test My Shields!, 35
Shift, 287
Simetría latente, 144
Sistema de defensa, 67
Sistema de facturación, 261
Sistema de identificación de virus, 267
Sistema de seguridad, 260
Sistema St-Cyr, 144
Slacker, 85
Smith, David L., 268
SMTP, 97
Sobreposición, 218
Sombrero negro, 46, 62
SpamHater, 54
Spider, 51
Spoofing, 58
SSL, 82,193
Stealh programming, 259
Subrutina, 232
Superordenador Cray, 171
Superposición (de un fotón), 215
Sustitución, 126, 166
Symantec, 97
System Tray, 120

T

Tabla de sustitución, 168
Tablas esqueleto, 144

- Tablón de mensajes, 266
- Tarjeta inteligente, 76
- Teclado, 110
 - Controladores de teclado, 118
 - Msconfig, 120
- Técnicas por la fuerza bruta, 61, 81
 - Ataques por la fuerza bruta, 153
 - Desciframiento por la fuerza bruta, 171
- Técnico antivirus, 266
- Teléfono celular, 42
- Terminología hacker, 249
- Texto de formato enriquecido, 283
- Texto original, 131, 227
- Texto plano, original, 125, 150, 227
- The Visual Basic Power Toolkit*, 127
- Tiempo de procesamiento del ordenador, 249
- Tonos C5, 42
- Transferencia de virus, 266
- Transmisión de fotones, 223
- Transposición, 126, 166
 - Transposición rotatoria, 168
- Trapdoor, 260
- Tribunal de Apelaciones de Estados Unidos, 253
- Túnel, 77

U

- Unión de Consumidores, 190
- UNIX, 262
- UPD, 94
- USA Today, 294

V

- Valor de cifrado, 181
- Valor numérico de un carácter, 155, 182
- Valores acumulativos, 236
- Variable clave, 232
- Variable i, 236
- Variable lk, 236
- Variable t, 232
- Variable v, 236
- VB Script File, 279
- Virus, 29, 49, 248
 - Bomba fork, 258

Bomba lógica, 255
Bomba temporal, 258
Caballo de Troya, 255, 262
Gran Gusano, 248
Gusano, 29, 255
Gusano benigno, 250
ILOVEYOU, 274
Melissa, 266
Visual Basic, 168, 228
 Lista de RND, 228
 RND, 228
Visual Basic para Aplicaciones, 269
 VBA, 282
Visual Basic Scripting, 278
Volcano, 259

W

WAN, 34
War dialer, 39
Warez, 40, 63
Whacker, 45, 59, 64, 85
Windows 2000 Profesional, 193
 Agente de recuperación, 197, 212
 Agregar o quitar complemento, 203
 Almacén de certificados personales, 211
 Archivo .pfx, 203
 Asistente para exportación de archivos, 207
 Asistente para importación de certificados, 211
 Atributos avanzados, 199
 Certificado, 194, 202
 Certificado autofirmado, 196
 Certificado-Usuario actual, 195, 206
 Cifrar contenido para proteger datos, 195
 Cifrar un archivo, 198
 Cifrar una carpeta, 201
 cipher/?, 202
 Copiar o mover archivos cifrados, 201
 Encrypting File System, 193
 Exportar archivo, 206
 Importar archivo, 203
 Mi cuenta de usuario, 205
 Microsoft Management Console, 203
 NTFS, 195
 Propiedades de documento, 199

Scripting Host, 278
Word, desactivar, 278
Word.exe, 302
Word, 147, 241
Seguridad, 148
Word Pad, 277
World Wide Consortium
hole, 260

27
otra a una clave, 156